

U.S.N.A. --- Trident Scholar project report; no. 329 (2005)

Direction of Arrival Estimation Using a Reconfigurable Array

by

Midshipman 1/c Danica L. Adams, Class of 2005
United States Naval Academy
Annapolis, Maryland

(signature)

Certification of Advisers Approval

Associate Professor Richard T. O'Brien
Weapons and Systems Engineering

(signature)

(date)

Associate Professor Kiriakos Kiriakidis
Weapons and Systems Engineering

(signature)

(date)

Acceptance for the Trident Scholar Committee

Professor Joyce E. Shade
Deputy Director of Research & Scholarship

(signature)

(date)

USNA-1531-2

Abstract

The goal of this project was to create a reconfigurable array that can determine the direction of arrival of a target. This goal was accomplished by using existing algorithms, in conjunction with redefining the assumed geometry of the array. These algorithms were modified to work with arrays that have the ability to move or change shape. The project investigated the effect of array rotation on the size of the data needed for the algorithm. It also examined the effect of changing the geometry from a purely linear array to an array that has two linear parts.

For demonstration purposes, ultrasonic sensors were used. Prior to implementing them, the proposed modifications to the geometry were simulated using a computer model. After the simulations were complete, the modifications were tested on the actual array. The first geometry examined with actual sensors was the linear array. The geometries investigated were those consisting of half of the array rotating such that the array formed an angle. These geometries were tested using the modifications made to the assumed geometry of the array within the algorithm. The modification of the assumed geometry allowed for different geometries to be tested.

This project correlates with research the Office of Naval Research is funding in non-conventional arrays. The results of this Trident project investigation led to further research and development of sonar arrays that may have practical applications for the Navy. An area of comparable research is that involving the resolution of ambiguities that occur when determining a direction of approach. The results obtained in this Trident project also fit into this area of research in both the Navy and the civilian world.

Keywords:

Direction-of-arrival

Estimation

MUSIC algorithm

Reconfigurable

Array

Experimental

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 074-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including g the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 6 May 2005		3. REPORT TYPE AND DATE COVERED
4. TITLE AND SUBTITLE Direction of arrival estimation using a reconfigurable array			5. FUNDING NUMBERS	
6. AUTHOR(S) Adams, Danica L. (Danica Lee), 1983-				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
US Naval Academy Annapolis, MD 21402			Trident Scholar project report no. 329 (2005)	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT This document has been approved for public release; its distribution is UNLIMITED.				12b. DISTRIBUTION CODE
13. ABSTRACT: The goal of this project was to create a reconfigurable array that can determine the direction of arrival of a target. This goal was accomplished by using existing algorithms, in conjunction with redefining the assumed geometry of the array. These algorithms were modified to work with arrays that have the ability to move or change shape. The project investigated the effect of array rotation on the size of the data needed for the algorithm. It also examined the effect of changing the geometry from a purely linear array to an array that has two linear parts. For demonstration purposes, ultrasonic sensors were used. Prior to implementing them, the proposed modifications to the geometry were simulated using a computer model. After the simulations were complete, the modifications were tested on the actual array. The first geometry examined with actual sensors was the linear array. The geometries investigated were those consisting of half of the array rotating such that the array formed an angle. These geometries were tested using the modifications made to the assumed geometry of the array within the algorithm. The modification of the assumed geometry allowed for different geometries to be tested. This project correlates with research the Office of Naval Research is funding in nonconventional arrays. The results of this Trident project investigation led to further research and development of sonar arrays that may have practical applications for the Navy. An area of comparable research is that involving the resolution of ambiguities that occur when determining a direction of approach. The results obtained in this Trident project also fit into this area of research in both the Navy and the civilian world.				
14. SUBJECT TERMS: Direction-of-arrival ; Estimation ; MUSIC algorithm ; Reconfigurable ; Array ; Experimental			15. NUMBER OF PAGES 67	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT		20. LIMITATION OF ABSTRACT

Acknowledgements

I would like to acknowledge all of the help I have received during this project. I want to thank both Mr. Joe Bradshaw and Mr. Norm Tyson from the systems support staff for all the help they have given me with my lab equipment and supplies. Mr. Joe Bradshaw has been an amazing resource and help with all of the circuitry in my project. His expertise in ultrasonic sensors contributed much of the success of this project. Thank you to Mr. Roy Goddard and Mr. Tom Price in the machine shop for their help in completing the design and building of the sonar platform. I also would like to thank both of my advisors, Professor Kiriakos Kiriakidis and Professor Richard T. O'Brien for all of their support and assistance with all aspects of the project. In addition, my thanks to the Trident Committee members, especially Professor Sarah Mouring, Professor John Burkhardt, and Professor Arthur Rachwald for their support and suggestions on presenting my material to others. I also want to thank my family and friends for putting up with me throughout this process and for all of their help. Finally, I want to thank Professor Joyce Shade for all the work that she does and her help throughout this process.

Table of Contents

Abstract	1
Acknowledgements	2
Table of Contents	3
Table of Figures	4
Table of Tables	4
1.0 Introduction	5
1.1 Problem Statement	5
1.2 Angle of Arrival of Planar Acoustic Waves	7
1.3 Data Collection	7
1.4 Frequency Content of Impinging Waves	9
1.5 MuSiC Algorithm	11
2.0 Apparatus Design	12
2.1 Original Concept	13
2.2 1 st Iteration	14
2.3 2 nd Iteration	17
2.4 3 rd Iteration	19
3.0 Simulations	21
3.1 Rotation Simulations	23
3.2 Reconfiguration Simulations	25
3.3 Laboratory Simulations	28
4.0 Experiments	29
4.1 Sensor and circuit testing	32
4.2 Initial array testing	33
4.3 Interim array testing	35
4.4 Field of View Testing	38
4.5 Final Platform Stationary Testing	39
4.6 Reconfiguration Testing	41
5.0 Simulations vs. Experiments	45
6.0 Conclusions	46
7.0 Further Research	47
Bibliography	48
Appendix A: Glossary of Terms	49
Appendix B: Circuit Schematics	50
Appendix C: Program code	52
Appendix D: Simulation Results	57
Appendix E: Experimental Results	60

Table of Figures

Figure 1: Direction of Arrival	6
Figure 2: Wave front Impinging on the Array	7
Figure 3: Analog signal	8
Figure 4: Digital signal.....	9
Figure 5: FFT sample output.....	10
Figure 6: Initial Concept of Reconfigurable Array with Three Mobile Sensors.....	13
Figure 7: RugBat™ Sonar Sensor.....	14
Figure 8: Original breadboard of sonar amplifier circuit	15
Figure 9: Linear to new configuration.....	15
Figure 10: Final Array Concept	16
Figure 11: Initial Array Setup	17
Figure 12: Interim array setup.....	18
Figure 13: Single Pulse Results (broadside)	20
Figure 14: Final Array Platform.....	21
Figure 15: Array Geometry within Simulations.....	23
Figure 16: MuSiC estimation before rotation (DOA 50° and $\theta_{nf}=135^\circ$).....	25
Figure 17: MuSiC estimation after rotation (DOA 50° and $\theta_{nf}=135^\circ$).....	25
Figure 18: Experimental Setup.....	26
Figure 19: Changing the Geometry of the Array	27
Figure 20: Flow chart of experiment.....	31
Figure 21: Directionality of Sensor.....	32
Figure 22: Single pulse (left side of array).....	34
Figure 23: Single Pulse (right side of array)	35
Figure 24: Quanser Model.....	36
Figure 25: MuSiC Results for Right Source – Trial 2	40
Figure 26: MuSiC Results for Left Source – Trial 1.....	40
Figure 27: MuSiC results for Right Source ($\theta_a=30^\circ$) – Trial 4	42
Figure 28: MuSiC Results for Right Source ($\theta_a=45^\circ$) – Trial 2.....	43
Figure 29: MuSiC Results for Right Source ($\theta_a=60^\circ$) – Trial 2.....	43
Figure 30: MuSiC Results from Left Source ($\theta_a=30^\circ$) – Trial 4.....	44
Figure 31: MuSiC Results from Left Source ($\theta_a=45^\circ$) – Trial 1	44
Figure 32: MuSiC Results for Left Source ($\theta_a=60^\circ$) – Trial 1	45

Table of Tables

Table 1: Change in Angles (Degrees) Due to Rotation.....	24
Table 2: Simulation results (DOA 40° and $\theta_{nf}=120^\circ$).....	28
Table 3: Simulation results (DOA 160° and $\theta_{nf}=70^\circ$).....	28
Table 4: Laboratory Simulation Results.....	29
Table 5: Pulse Width (sec) of Single Pulse.....	33
Table 6: Range of Array Sensors	38

1.0 Introduction

The ability to track targets is one of the major concerns of the United States Navy. In order to detect and track targets underwater, sonar (SOund NAvigation Ranging) is the main sensor used. The Navy uses two sonar systems: the Sound Surveillance System (SOSUS) and the Deployable Autonomous Distributed System (DADS) [ONR, 2004]. These systems utilize passive sonar arrays that are fixed in position to detect submarines. Some experimentation with the organization of arrays is ongoing currently [Walker, 2003]. In addition, the Navy uses towed arrays in linear, spherical, or cylindrical geometries. The cylindrical and spherical arrays are found in the sonar dome of a ship or a submarine. The information from these arrays is then processed using a variety of methods [Horton, 1969; Urick, 1975; Ziomek, 1985].

This project is building on the idea of sonar systems. However, instead of using fixed position sensors, the effects of using a reconfigurable array were examined. Would the ability to reconfigure the array reduce some of the maneuvers necessary to resolve ambiguities? The ability to change the geometry of the array also allows the improvement of the MuSiC (Multiple Signal Classification) algorithm when a signal and a reflection both exist. The MuSiC algorithm is used to determine the direction of arrival. In order to examine the effects of changing the geometry, both computer simulations and physical experiments were utilized. The parameters of the simulation were kept constant with those of the lab experiments, in order to compare the results. For these experiments, an array of five ultrasonic sensors was used.

1.1 Problem Statement

The main purpose of this project was to modify the MuSiC algorithm to calculate the direction-of-arrival (DOA) using a reconfigurable array (see Figure 1). In order to calculate the

DOA, sensors were needed to obtain information about the environment. The project took various physical principles into account to estimate the direction of arrival. Sound travels in planar waves when the source is far from the receiving sensors. These waves would hit each sensor at slightly different times. Theoretically, this time delay can be determined from the Fast Fourier Transform (FFT) of signals received at the sensors. Practically, the FFT data is inputted into the MuSiC algorithm, which computes the estimated angle of the source.

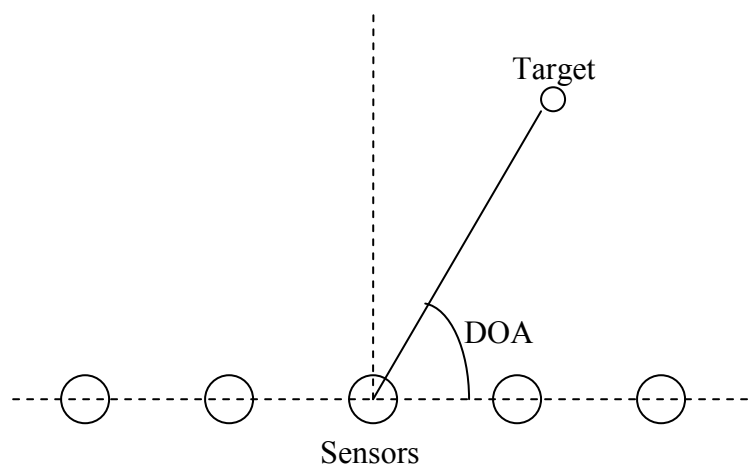


Figure 1: Direction of Arrival

The project examined the DOA estimation when a near field source was present and when it was just the far field source. The near field source represented the situation in which the signal from the far field source is reflected off of an obstruction and arrives at the sensors from a different angle. In this case, this reflected signal is considered noise that is related to the far field signal. The near field source does not travel as a straight plane but rather as a spherical plane from a point source. Simulations were done to look at the effect of this near field source would have on estimating the DOA of the far field source. In the experiments, only the far field source was present. Both situations were examined to determine the effectiveness of the modifications.

1.2 Angle of Arrival of Planar Acoustic Waves

The primary concept for identifying the DOA is the time delay between the receipt of the same wave front at each sensor (see Figure 2). Sound waves travel as planar waves that will come in contact with each sensor in sequence. As sound waves are received, each sensor of the array will detect them at slightly different times. The time delay is given by the expression

Equation 1

$$\tau = \frac{d \cos(\theta)}{c}$$

where τ is the time delay, d is the spacing of the sensors, c is the propagation speed and θ is the direction of arrival. If the signal is from an angle broadside to the array, there will be no time delay.

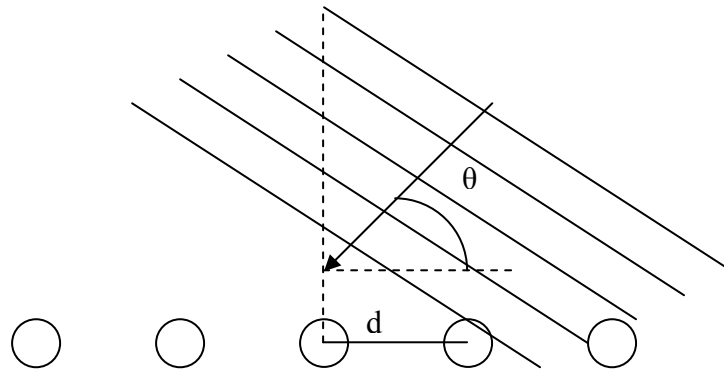


Figure 2: Wave front Impinging on the Array

1.3 Data Collection

The transmitter emitted an ultrasonic signal that was detected by the sensors. The detection was in the form of a voltage difference across the sensor which was sent through an amplifying circuit and comparator circuit. The data collected from the sensors is in the form of analog voltage (Figure 3). In order to change the analog signal into a digital signal, the signal is processed through a comparator circuit (see Appendix B). This circuit compares an input

voltage to a threshold value of 0.20V. Once the signal rose above the threshold, the output of the circuit was 5V; otherwise it was zero volts (Figure 4). The analog signal produced by the sonar amplifier circuit is a periodic signal related to the period of the transmitter and has a time constant related to the resistive load in parallel with the final capacitor. This time constant, $\tau = RC = (100k\Omega)(.01\mu F) = .001s$ affected the time required to return to the steady state value of zero after the sensors received an input.

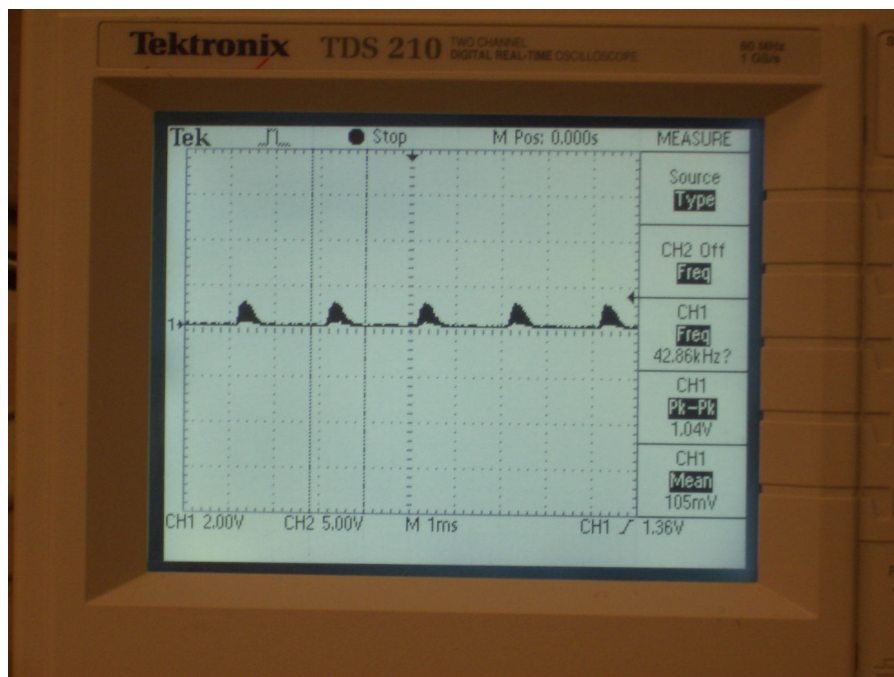


Figure 3: Analog signal

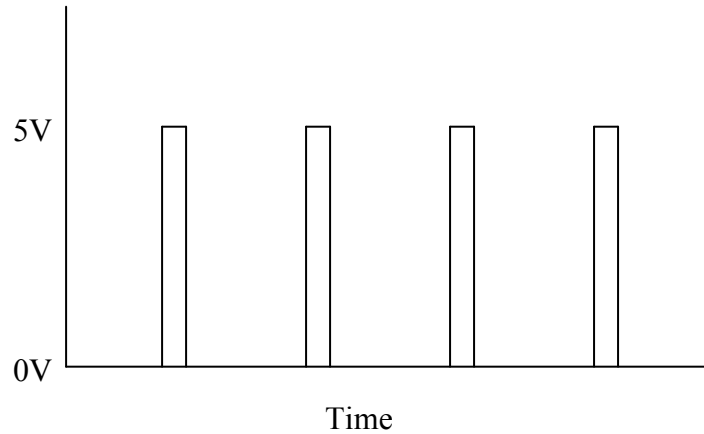


Figure 4: Digital signal

1.4 Frequency Content of Impinging Waves

In order to process the data into a form that could be implemented into the MuSiC algorithm, a FFT is performed. The FFT is an algorithm that reduces the number of computations needed to calculate the Discrete Fourier Transform for n points from $2n^2$ to $2n\lg(n)$ where \lg is log base two. The DFT reveals periodicities in input data as well as relative strengths of any periodic components [Weisstein, 2005]. The sequence of n complex numbers x_0, \dots, x_{n-1} are transformed into the sequence of n complex numbers f_0, \dots, f_{n-1} by the DFT according to the formula:

Equation 2

$$f_j = \sum_{k=0}^{n-1} x_k e^{\frac{-2\pi i}{n} jk} \quad j = 0, \dots, n-1.$$

where e is the base of the natural logarithm and i is equal to the square root of -1 [Wikipedia, 2005]. Within the frequency domain, the magnitude and phase are determined (Figure 5). The FFT converts the information from the spatial domain to the frequency domain. In order to obtain the information needed to determine the DOA, the maximum magnitude is found. This magnitude is located at the frequency of the source signal. The magnitude and phase of this

point is stored in a matrix for further processing within MuSiC. The FFT works best when the number of points (n) used is a power of two. Within this project, groups of $n=512$ points and $n=1024$ points were used. Each group of points represented a snapshot of data. The FFT was performed on each of these groups for all five sensors. The magnitude and phase information from the source's frequency are stored in a matrix. This matrix contains five rows (one for each sensor) whose length depends on the number of snapshots that were processed from the raw data. The complex numbers that were stored in the matrix contained information about the strength of the signal as well as the time delay information within the phase.

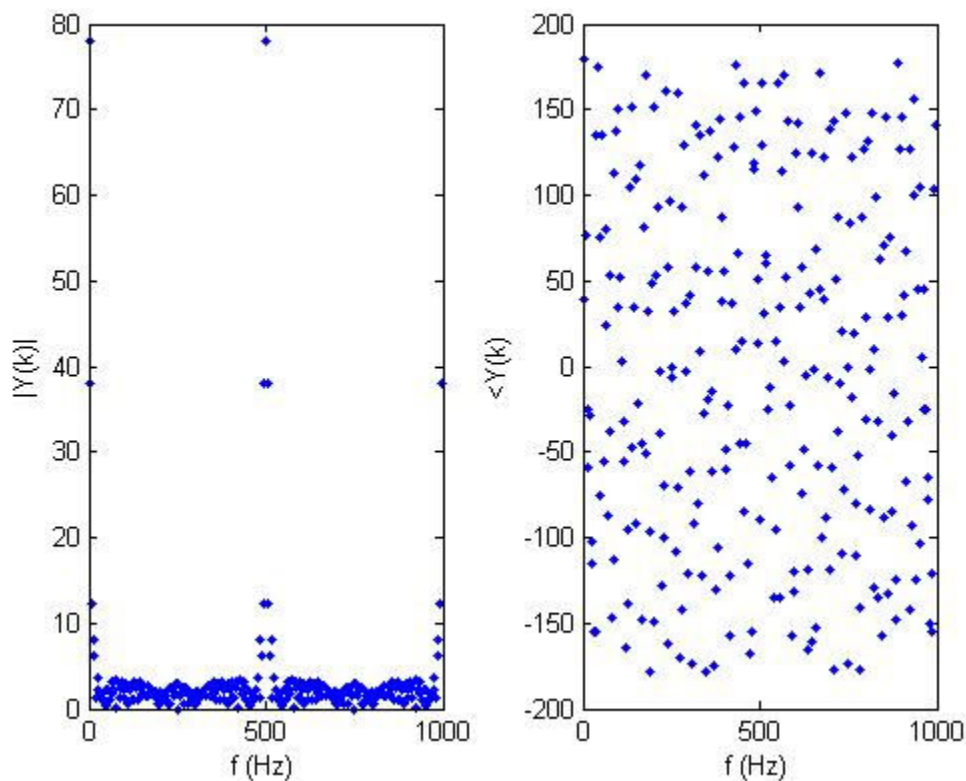


Figure 5: FFT sample output

Within the script used to perform the FFT, multiple changes were made throughout the project. The major change was in defining the data matrix that would be exported into the

MuSiC algorithm. This matrix contains the magnitude and phase information of the maximum value of the FFT. In the original setup, it was assumed that each sensor would have this maximum at approximately the same spot in the FFT. However, it did not take into account the case when the first sensor may not receive a signal at all. In this case, its value would be the same throughout and it would define the maximum with a zero angle. The original setup used only the index from the first sensor to define the data matrix rather than take each sensor separately. This caused the data matrix to contain the wrong information. Instead of containing magnitude and phase information from the frequency of the signal, it contained the magnitude and phase information when the frequency was zero. In this case, there was no phase information. All of the data points were real numbers as opposed to the complex numbers in polar form. The script was changed to incorporate the fact that the index of the maximum magnitude would vary from sensor to sensor. Therefore each sensor was looked at independently to determine the maximum magnitude and corresponding phase information.

1.5 MuSiC Algorithm

The signal processing is implemented to determine the DOA of a contact. The algorithm uses data from the sensors and determines the DOA. In order to process the data, the direction finding method known as the MuSiC algorithm was used [Allen, 1991]. This algorithm was developed to analyze multiple input signals and to find the direction of an acoustic wave impinging on an array of sensors. The MuSiC algorithm uses the magnitude and phase data from the Fast Fourier Transform. The signal received by the sensor array is:

Equation 3

$$\begin{array}{ccccc} X & = & A(\theta) F & + & W \\ \text{(Signal)} & & \text{(Modeled)} & & \text{(Unmodeled)} \end{array}$$

where X is the received signal, $A(\theta)$ is the array model and a function of the unknown DOA, θ , F is the source model and W is the white noise within the system. MuSiC processes the data matrix XX^T assuming that F and W are uncorrelated. MuSiC uses eigenvalue decomposition to detect the frequencies within a signal and create a noise spectrum with a peak at the DOA.

When using this algorithm, there are a number of assumptions that must be met [Hassab, 1989]. These include the assumption that the signal and sensor noise are uncorrelated and that the signals are uncorrelated. It also assumes omni-directional sensors that are fixed in location. The acoustic waves are assumed to be plane waves and have a constant propagation speed. Many of these assumptions are examined for their validity throughout this experiment.

The algorithm was written as a MATLAB program and was implemented using a MATLAB interface with Quanser [Quanser, Inc., 2005]. This allowed the digital input to be read into the computer for processing. Real and simulated data have been used to test and calibrate the algorithm.

2.0 Apparatus Design

The apparatus design evolved through several iterations. The primary objective was to construct a reconfigurable array. The method of reconfiguring the sensors was the main part of the design that was changed throughout the project. The circuitry was also modified to accomplish the desired requirements.

2.1 Original Concept

The original plan involved using three Sonar-equipped robots as the mobile platform (Figure 6). The robots would provide a mobile platform that would reconfigure the array. RugWarrior™ robots were chosen because they had attachable sonar sensors (Figure 7) [Jones, 1992]. This idea was abandoned in the beginning of the research for a couple of reasons. The primary reason was that the initial sensors accompanying the RugWarriors™ were Polaroid sensors that were set to work as active sonar. The sonar sensors that would have been used were both transmitters and receivers requiring a large amount of power as well as having to switch from transmitting to receiving. The sensors would have required significant alterations in order to be in the receiving mode that is required for this application. For this project, passive sonar was required. Therefore, a new sensor was found that could be set into purely listening mode. Another reason for moving away from the robots was the ability to increase the number of sensors in the array. The cost of the robots limited the number of arrays that could be used. The robots also added an additional source of complexity that did not directly contribute to the primary objective of the project.

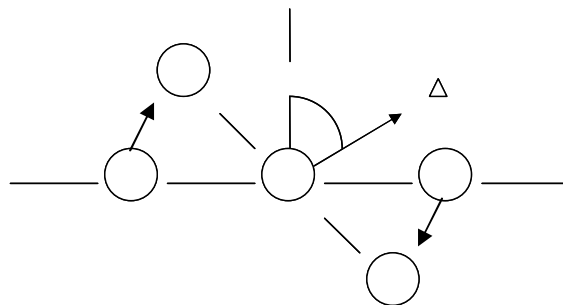


Figure 6: Initial Concept of Reconfigurable Array with Three Mobile Sensors

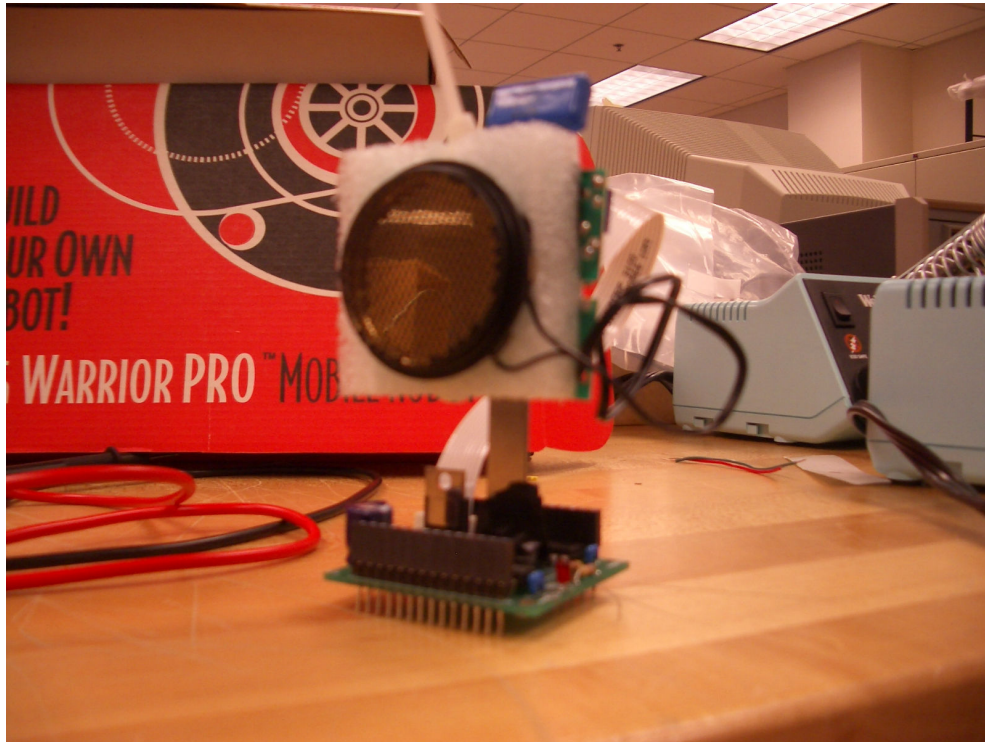


Figure 7: RugBat™ Sonar Sensor

2.2 1st Iteration

An array of five ceramic ultrasonic sensors was developed to replace the mobile robots. These sensors were chosen due to their availability, ability to perform the required function, and reduced cost. The first challenge was to design and build a circuit that could take the analog sensor output and convert it into a digital signal. This circuit was divided into two parts; a sonar amplification circuit and an A/D converter circuit (see Appendix B). Both parts of the circuits were laid out on breadboards for the initial testing. Later in the project, the amplifying circuit was made on printed circuit boards. These boards were smaller than the original boards (seen in Figure 8). Since each board had two circuits, only three boards were needed instead of five. This also provided an extra circuit as a backup.

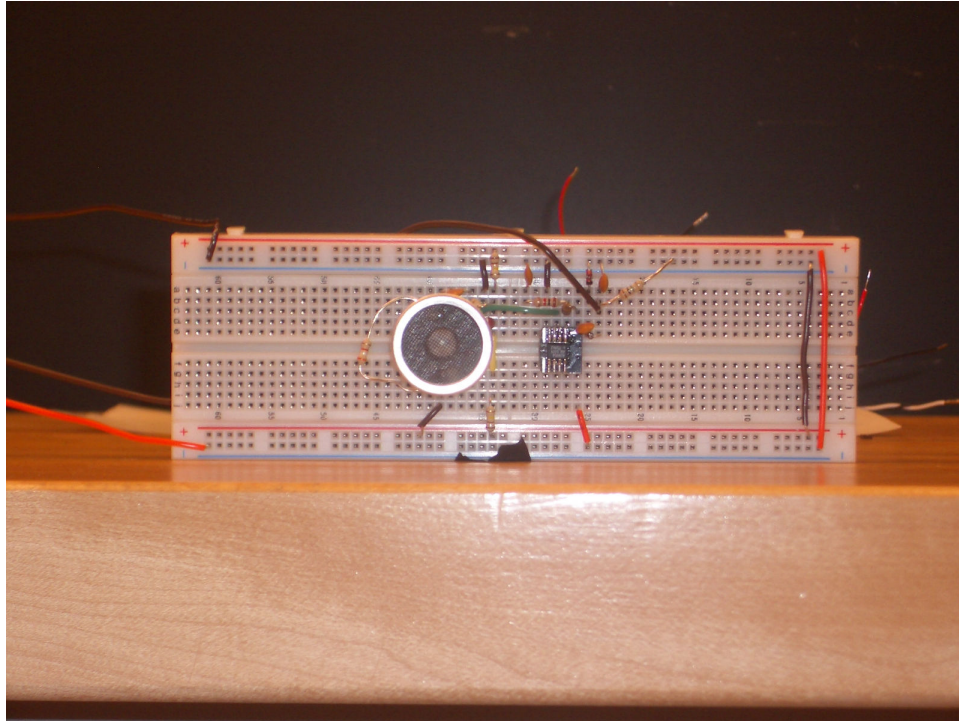


Figure 8: Original breadboard of sonar amplifier circuit

After the circuit was built, a new design for the array was needed. For the purpose of this project, the array should be able to go from a linear array to one with an angle between two parts (

Figure 9). No other geometries were tested or examined. Therefore a platform with the ability to rotate two different arms was necessary. This was accomplished through the use of two separate servomotors to move each arm individually. Each arm had two sensors attached to it and the fifth sensor was between the two servomotors (Figure 10).

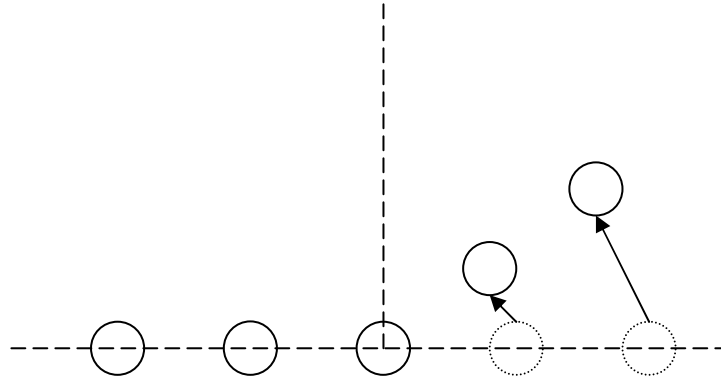


Figure 9: Linear to new configuration

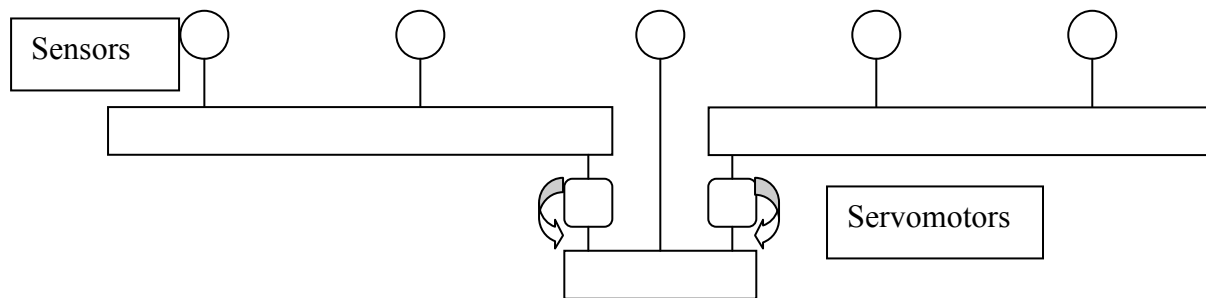


Figure 10: Final Array Concept

The other aspect of the design that needed to be considered was the spacing of the array. The spacing was limited by physical size and the frequency of the signal. The desired spacing of the array would be at half of the wavelength of the signal. The optimum spacing is related to the Nyquist frequency, which is equal to twice the frequency of the signal [Weisstein, 1999]. This is the frequency at which a signal can be sampled to reconstruct the original signal. If the signal is sampled at a higher rate than this frequency, no additional information is gained. However, additional computations are needed to sample at a higher frequency. A smaller frequency than the Nyquist frequency will result in the loss of information. Since the frequency is inversely related to the wavelength, in order to sample at twice the frequency, the spacing needs to be half of the wavelength of the signal.

Therefore the frequency of the signal is related to the spacing of the array. On one hand, if the frequency were too small, it would cause the array spacing to be extremely large. This spacing would not be practical in the lab. On the other hand, the frequency of the signal could not be faster than the sampling rate of the computer. Increasing the frequency reduces the size of the array. Therefore physical and computational limitations existed in determining the spacing between elements of the array. The best frequency would be the largest that could be sampled without missing any data. The sampling rate of the Quanser interface is 2 kHz. This led to using a signal with a 500 Hz frequency to carry the bursts of 40 kHz from the ultrasonic transmitter. As a result of using a 500 Hz frequency, the spacing between the elements was a little over 1ft. The spacing was determined using the following calculations:

Equation 4

$$\lambda = \frac{c}{f} = \frac{343m/s}{500Hz} = .646m \qquad d = \frac{1}{2}\lambda = \frac{1}{2}(.646m) = .323m = 1.14ft$$

λ = wavelength of signal

c = speed of sound in air

f = frequency of signal

d = distance between sensors

2.3 2nd Iteration

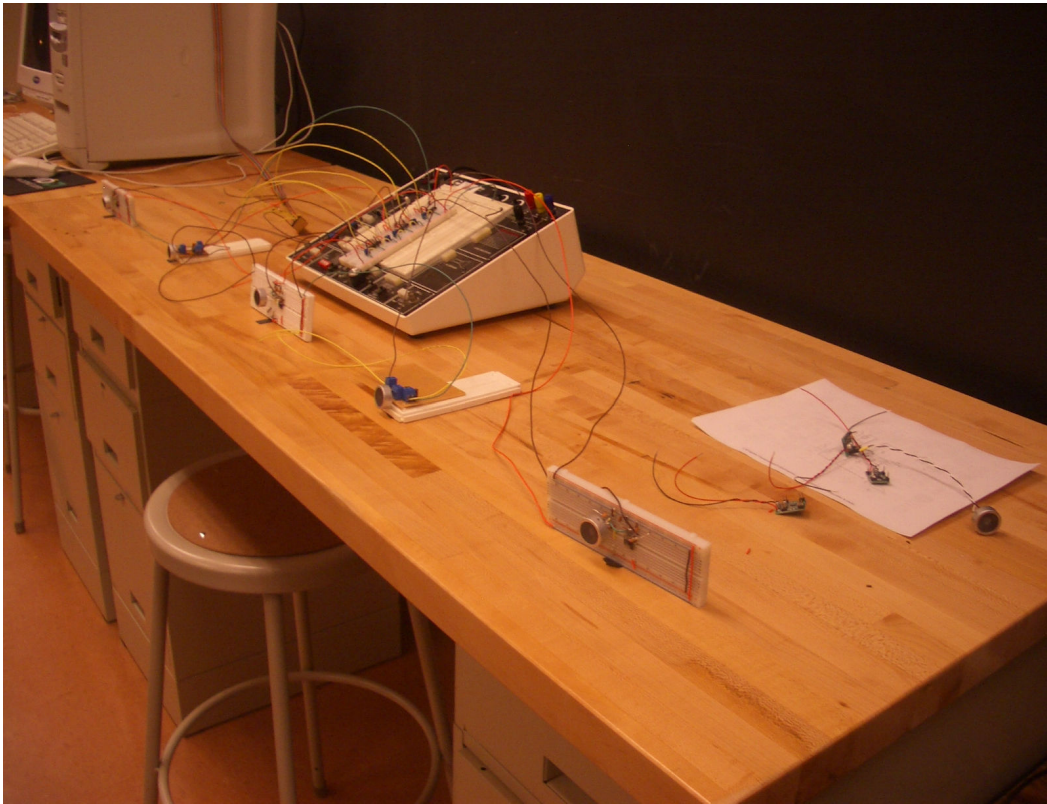


Figure 11: Initial Array Setup

During the initial testing stages, a Devantech transmitter/receiver was used as the transmitter [Coe, 2001]. The array was set up with the sensors on the breadboards (Figure 11) in order to test inputting data from the array into a computer. The transmitter was connected to a switch in order to control the pulse emitted. After the first few initial tests, the array was modified to use the printed boards that had been purchased. In order to do so, stands were necessary to hold the sensors in place. Two sensors shared the same board, so they were connected with long wires. The stands that were used had alligator clips that clipped onto the wires that connected the sensor to the circuit board (Figure 12).

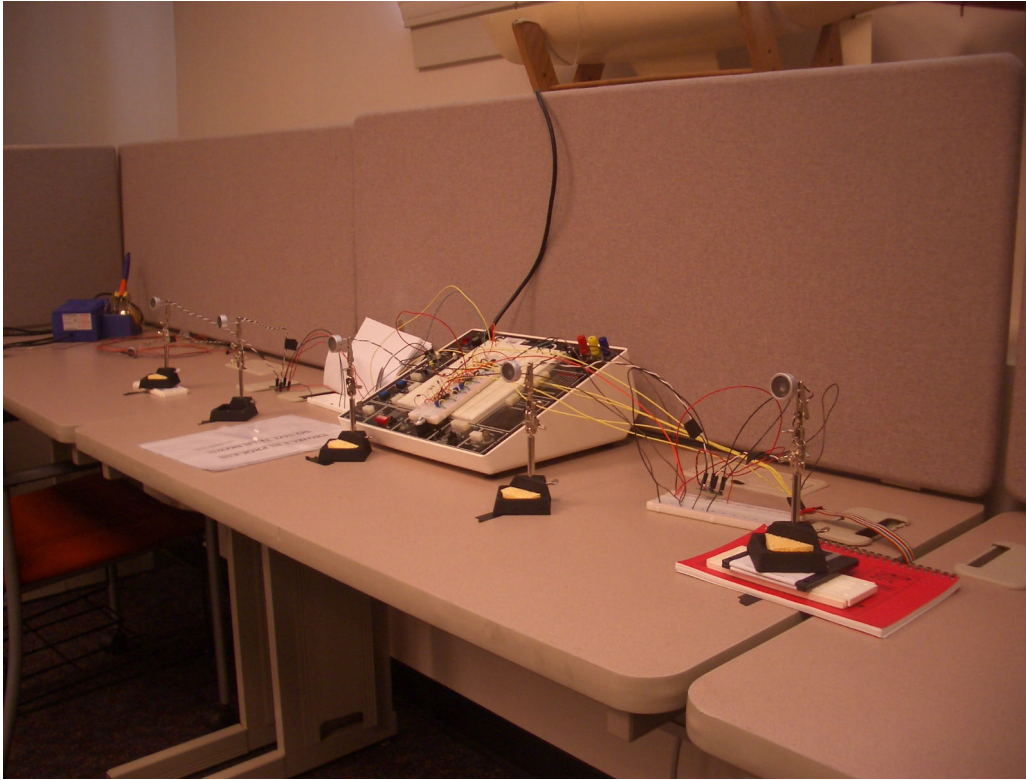


Figure 12: Interim array setup

This array was used for further testing with single pulses as well as testing with a continuous signal. The printed circuit boards used resistors and capacitors that had greater precision. The greater precision reduced the noise within the circuits. The precision is related to the uncertainty in the values of the resistors and capacitors. This allowed for a better precision.

2.4 3rd Iteration

The Devantech transmitter/receiver worked while testing with single pulses (Figure 13). However, when it was attached to a function generator to create a 500 Hz signal, there were limitations. The first problem that had to be solved was that the time constant of the circuit boards was different than it had been in the initial tests (Figure 13), which created much longer pulses. The size of the capacitor in the sonar amplifier circuit was reduced when the printed circuit boards were made. This required a larger resistor to be placed across the capacitor as a

load to achieve the same time constant. After this was solved, the frequency of the signal that was sent was still only about 100 Hz even when the function generator was at 500 Hz. It was determined that the transmitter is designed to have a 10ms pause between pulses. The Devantech is designed to be a range finder and therefore has a built in wait time to listen for an echo. As a result, a new circuit and transmitter were used (see Appendix B). Initially a logic level gate and an AND gate were used to create the 40 kHz signal to send out at a frequency of 500 Hz. However this created a longer burst than was previously used. To correct the length of the 40 kHz burst, a function generator that had the ability to control the duty cycle was chosen. The logic level gate was abandoned and the circuit was changed to model that of the Devantech transmitter. The only difference is that the trigger pulse and the frequency of the signal could now be controlled. The pause that existed in the Devantech was no longer an issue. The transmitter could now be used to emit a continuous signal for array testing.

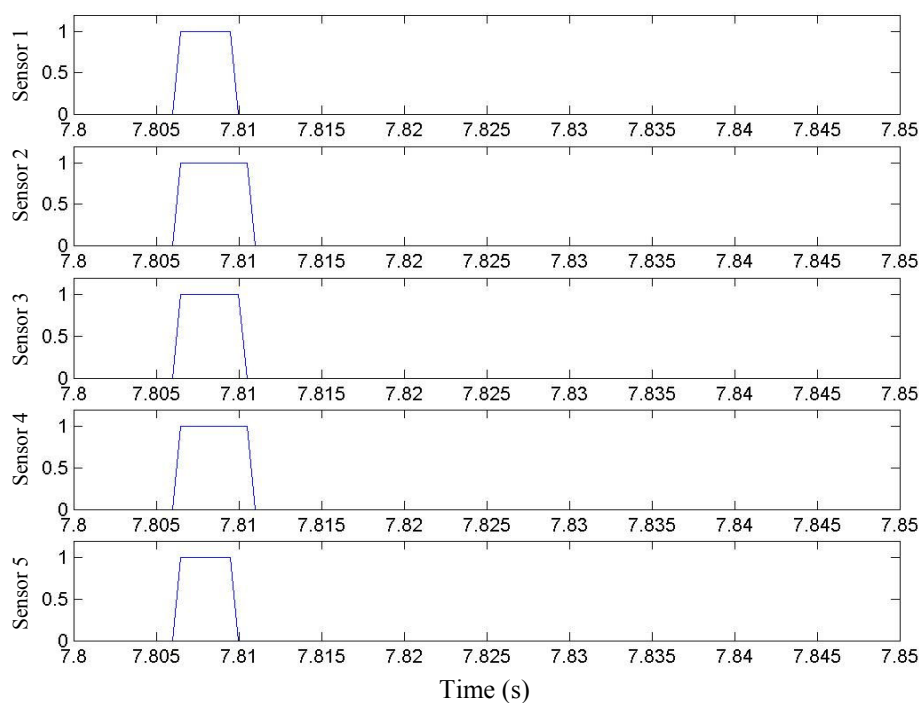


Figure 13: Single Pulse Results (broadside)

The final step in the array design was building the platform to make the array reconfigurable. Two large servomotors were used to turn each individual arm. The arms were made from plywood. Each sensor was mounted on a smaller servomotor and then attached to the arms of the platform. The center sensor and the two large servomotors were mounted on a base made from plywood as well. In order to prevent the arms from bending too much from their own weight, long supports were made from plastic (Figure 14). This allowed the arms to slide on them while they rotated. Since real time motion is not a priority in this project, friction was not an issue. The small servomotors allowed for the sensors to have an additional degree of freedom to the movement of the entire side of the array. The final design was a result of many changes and modifications to the original idea.



Figure 14: Final Array Platform

3.0 Simulations

The simulation models a far field acoustic source where the speed of sound, separation of the array, direction of arrival, and frequency of the signal can be specified. This far field model assumes that the source is an infinite distance away from the array. For practical purposes this

distance was assumed to equal $d_{ff} = \frac{N^2 \lambda}{4}$ where N is the number of sensors and λ is the wavelength of the signal. The calculated ideal distance was

Equation 5

$$d_{ff} = \frac{(5)^2 (.646m)}{4} = 4.29m = 14.3ft .$$

The actual distance used throughout the experiments was 10 ft.

The far field signal was modeled using the parameters that would exist in the lab. Some of the parameters that were defined in the simulations included the speed of sound, frequency of the signal and number of sensors. The speed of sound depends on the medium, assumed to be air in order to reflect the results when the practical experiments were performed in the lab. The speed of sound in air is 343 m/s. The spacing of the array depends on the frequency of the signal and the speed of sound. The optimum spacing is half of the wavelength of the signal, $\lambda = c/f$, where c is the speed of sound and f is the frequency of the signal. The number of sensors was also defined within this code as five.

In each of the simulations, the effect of a reflection signal was observed. A near field signal was created to simulate this reflection signal. The near field signal has similar parameters as the far field signal. The distance from the array was an additional parameter in the near field signal. In the simulations, the MuSiC algorithm assumed a far field model. Within the algorithm the two signals were added together prior to the processing. The X matrix from both sources were added together to create the X matrix that was read into the MuSiC algorithm (see Equation 3). The effects caused by rotation, various reflection angles and different geometries in these simulation experiments were observed. The purpose of doing the simulations is to demonstrate the effect of the reconfigurations of the array (Figure 15). Within these simulations, the near field source was assumed to be a reflection of the far field source. The near field source had the same frequency as the far field source. Since it is a reflection, the signal strength of the near field source was assumed to be one tenth of the strength of the far field source within the simulations.

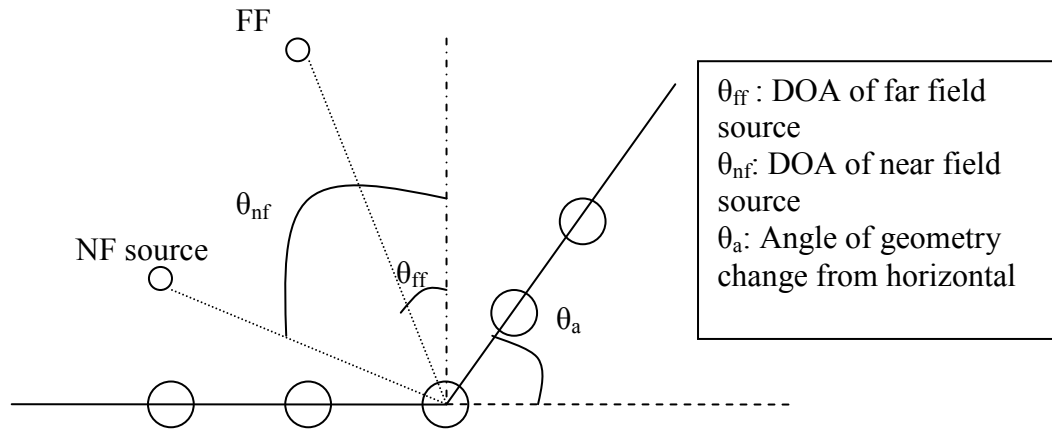


Figure 15: Array Geometry within Simulations

3.1 Rotation Simulations

The first set of simulations examined the effect of rotating the array. They were done to determine whether rotation alone could improve the results of the algorithm. If so, rotation can be accomplished electronically through beam forming, which does not require the array to be physically moved. Within these experiments, both the far field signal and the near field signal were taken into account. The simulations looked at two different near field reflection angles. Within the simulations, the array was rotated to put the reflected signal at the end of the array. The angle inputted in to the simulation was changed in order to accomplish this (Table 1).

Table 1: Change in Angles (Degrees) Due to Rotation

DOA (θ_{ff})	θ_{nf} (reflection)	New DOA	New θ_{nf}
10	80	110	180
30	80	130	180
50	80	150	180
70	80	170	180
90	80	10	0
110	80	30	0
130	80	50	0
150	80	70	0
170	80	90	0
10	135	55	180
30	135	75	180
50	135	95	180
70	135	115	180
90	135	135	180
110	135	155	180
130	135	175	180
150	135	15	0
170	135	35	0

Once all of the simulations were completed, the magnitude of the MUSIC estimation before and after rotation was compared (Figure 16 and Figure 17). Within these experiments, there was no significant improvement due to the rotation. The rotation improves the results when the near field signal and the far field signal were close to each other, which is an unlikely scenario. Otherwise, the magnitude of the MUSIC estimation was greater prior to rotation (see Appendix E). Therefore, since rotation did not improve the estimation, other changes in geometry were examined. Rotation is not the only option of changing the array. Beam forming would only accomplish the rotation of the array and therefore is not effective to accomplish the other geometries proposed. Reconfiguration was still an option to improve the estimation of the direction of arrival.

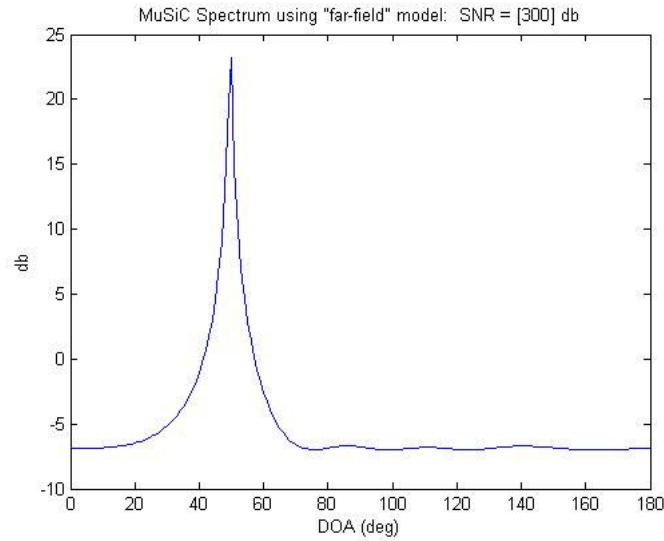


Figure 16: MuSiC estimation before rotation (DOA 50° and $\theta_{nr}=135^\circ$)

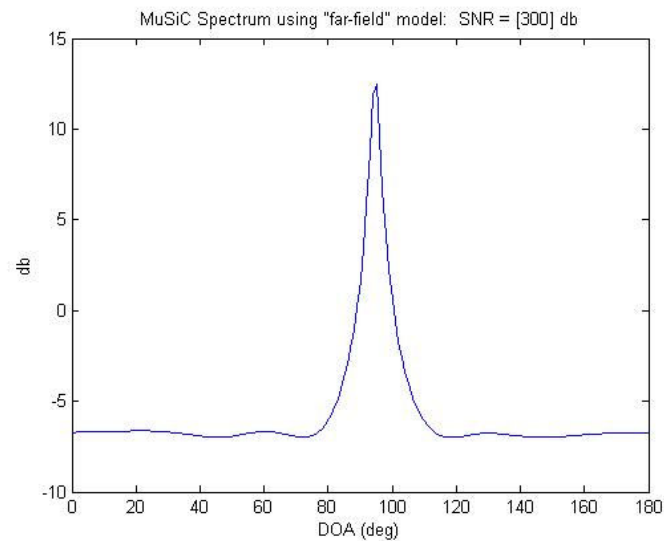


Figure 17: MuSiC estimation after rotation (DOA 50° and $\theta_{nr}=135^\circ$)

3.2 Reconfiguration Simulations

After examining the effects of simply rotating the array, the simulations began to include geometries other than a linear array. The main change in geometry consisted of varying the angle of one side of the array. In order to reconfigure the array within the simulations, modifications had to be made to the script that defined the array's geometry.

For the purpose of the simulations and experiments, the sensors in the array were numbered as seen in Figure 18. The angles were defined based on the perspective from the array. When discussing the array, the left hand side of the array was defined to consist of sensors four and five. The angles were defined according to this specification. These definitions were used throughout the experiments.

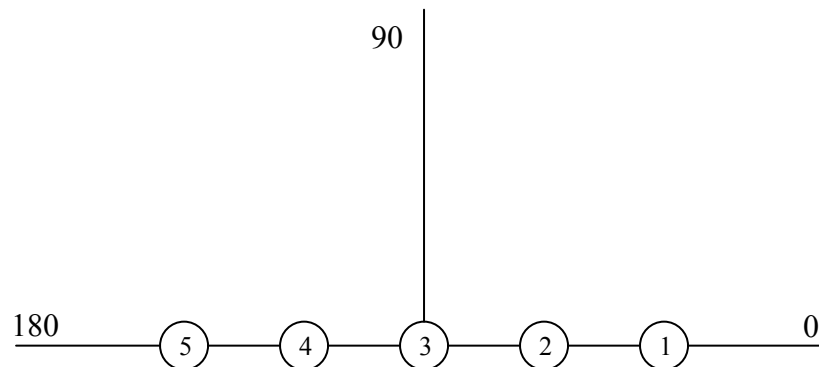


Figure 18: Experimental Setup

The array geometry was defined in terms of the angle the arm made with the horizontal. Three element vectors defined the geometry of the array. Each part of the vector defines the position of the sensor in x, y and z components. For all of the experiments and simulations, the z component was defined to be zero. Everything was assumed to be a 2-dimensional problem. When the array was linear, the x component consisted of the unit vector multiplied by the spacing distance. Within the definition of the array geometry, the sensors are spaced between $-2d$ and $2d$ (d being the spacing between the elements). In order to reconfigure the array, the vectors were redefined to reflect the new positions of the sensors. Depending on which side of the array was moved, two of the sensors would have both x and y components instead of being along a straight line with the other three sensors. The x component was defined in terms of the cosine of the angle and the y component was defined in terms of the sine of the angle (Figure 19).

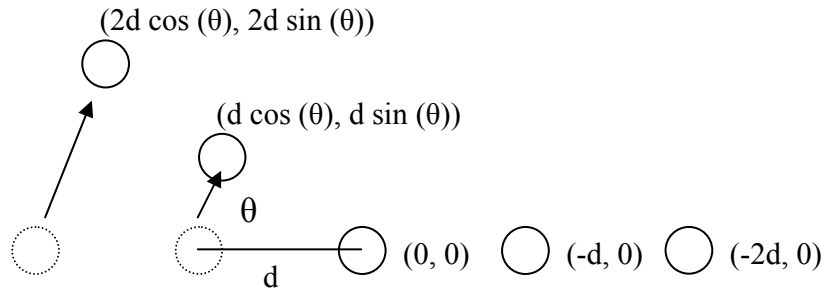


Figure 19: Changing the Geometry of the Array

Within these simulations, two different directions of arrivals were used: 40° and 160° . For each different angle, the reflection angle was varied from zero to 180° in increments of 10° . However, some of these reflection angles are not realistically possible due to basic geometric and physical principles. For example, a reflected signal that arrived from the same direction as the source would not be possible. It would be possible to have a near field and far field signal on the direction of arrival if the two sources were separate. For this project, a far field source and its reflection was the case of interest.

The angle of one side of the array was 30° , 45° , or 60° while the other side remained fixed. For each of the two directions of arrivals, the opposite side of the array was moved. This was to reflect how the array would change in application.

In order to determine the effectiveness of reconfiguration in the case where a reflected signal exists, the magnitudes and the estimated DOA of the MuSiC plots were compared. The results from the reconfigured array were compared with those from the linear array. In general, the reconfiguration of the left end of the array (Figure 15) resulted in higher magnitudes than the linear array (Table 2). The reconfiguration angle of 45° was better than the linear array about half of the time. The reconfiguration also tended to either estimate the DOA at or below the

actual DOA while the linear array was often above the actual DOA. The results from the reconfigured array were an improvement over the linear array a large percentage of the time.

Table 2: Simulation results (DOA 40° and $\theta_{nr}=120^\circ$)

	Linear Array	Left 30°	Left 45°	Left 60°
Magnitude	19.6 dB	21.2 dB	19.4 dB	23.3 dB
DOA estimation	42°	39°	37°	37°

The results from the reconfiguration of the right end of the array were slightly different. This was in part due to the fact that the DOA examined was 160°, which was much closer to the end of the array than the middle of the array. In this case, changes of 30° and 45° in the geometry had a greater magnitude than the linear array about half of the time. 60° did not improve the magnitude of the estimation at all. However, in this case, the actual estimation of the DOA was improved with the changes in geometry (Table 3). The linear array always estimated the DOA to be higher than it actually was. Each of the geometries was usually much closer to the actual DOA. The reconfiguration of the right end of the array also improved the estimation performance of the array.

Table 3: Simulation results (DOA 160° and $\theta_{nr}=70^\circ$)

	Linear Array	Right 30°	Right 45°	Right 60°
Magnitude	17.9 dB	21.9 dB	18.8 dB	15.5 dB
DOA estimation	164°	156°	160°	159°

3.3 Laboratory Simulations

The final simulations were performed to reflect the parameters that would exist in the lab. The simulations used the directions of arrival that were used in the reconfiguration testing of the array. These simulations did not reflect the case in which a second source is present. Only the

far field source was present in the simulations. The signal to noise ratio was assumed to be 0 dB within this part of the project. This was to reflect the noise that was present within the physical apparatus. Angles of 83° , 90° , and 95° were used for the directions of arrival in these simulations. All three angles were modeled with the linear array in order to obtain data with which to compare the experimental results. For these simulations, one arm of the platform remained horizontal while the other was varied between three angles (30° , 45° and 60°). When the source was on the left end of the array, the right arm of the apparatus was moved. Similarly, data was collected from the source at an angle to the right of broadside when the geometry of the left end of the apparatus was changed. The simulations did not model a reconfiguration of the array for the case when the source was broadside to the array. The magnitude of the peak of the MuSiC plot and the DOA estimation were compiled (Table 4).

Table 4: Laboratory Simulation Results

DOA 83				
	Linear array	Left 30	Left 45	Left 60
MuSiC peak	24.4 dB	18.76 dB	19.85 dB	22.17 dB
MuSiC estimation	83°	83°	83°	81°
DOA 95				
	Linear array	Right 30	Right 45	Right 60
MuSiC peak	22.6 dB	24.96 dB	20.37 dB	18.67 dB
MuSiC estimation	94°	95°	94°	94°

4.0 Experiments

After the simulations were completed, experiments with the physical array were begun. Each individual sensor and circuit needed to be tested prior to testing the direction of arrival algorithm with the array. The first experiments determined the directionality of the transmitters and receivers. The circuitry was tested in order to test their ability to achieve the type of output necessary for the calculations. Once the ranges and directionality of the sensors were

determined, each individual sensor was tested using a single pulse from the transmitter. Within these experiments, the width of the pulse was examined and compared, to calibrate the sensors to work within the parameters set for the array. The circuits were tested in the same manner as the sensors (see section 4.1 for results).

After the individual circuit and sensor testing, the array was tested as a whole. The initial tests contained only a single pulse. After the array was tested detecting a single pulse, a continuous signal or pulse train was produced. This signal was used in the rest of the experiments. The initial experiments using the pulse train focused on obtaining data to determine whether MuSiC would estimate an angle. The array was tested using three different angles (68.2° , 116.6° and 90°). Another aspect of these experiments was determining at which angles all five sensors detected the signal, varying the DOA from 68.2° to 116.6° . After MuSiC was shown to work with this array, further tests were run to experimentally show the effects of changing the geometry of the array. This included varying the array angle from 30° to 60° on either side of the array and testing an angle on the opposite side of the array from the geometry change. Within the experiments, the amount of data collected was examined. Another factor examined was how the FFT code packaged the data. Throughout the experiments, the data was collected and then processed with the FFT and MuSiC algorithm (Figure 20).

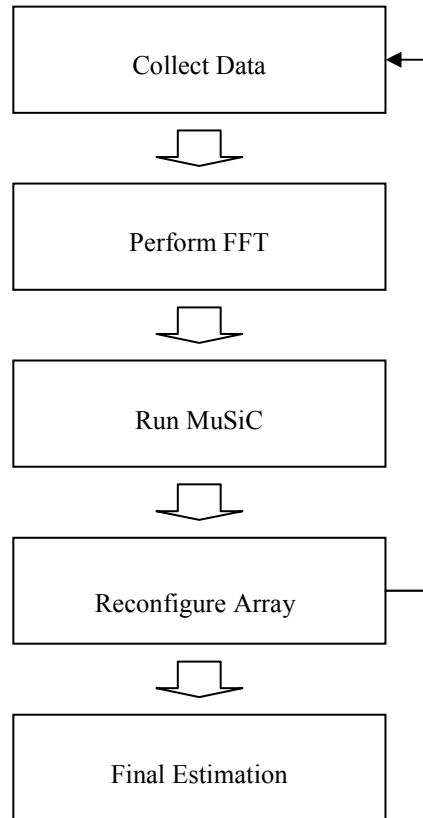


Figure 20: Flow chart of experiment

In the experiments, the source was located at a distance of 10 ft from the face of the array. At this distance, the far field assumption is reasonably valid for the performance of MuSiC. The ideal distance was calculated previously to be 14.3 ft. The discrepancy between the calculated ideal distance and the distanced used was due to the available space in the lab and the range of the sensors. The source was varied along a horizontal line at 10 ft from the array face to create different directions of arrival. If the distance had been increased to the calculated distance, greater distances along this line would have been required to achieve the same angles. This was not feasible in the space provided.

4.1 Sensor and circuit testing

The initial tests examined the sensors' range and directionality. Throughout these tests, changes in the circuitry were made. The first test investigated the range and the angle of the cone (Figure 21). A single receiver was used to determine the maximum range of the transmitter. In order to increase this range, an amplifier was added in the circuitry for each sensor. Each sensor has a cone of about 65° in which it can detect a signal. The maximum range measured was about 18ft. It was also determined that a resistor was needed across the final capacitor of the receiving circuit in order for its output to be amplified and then digitized for input into the computer. In steady state, capacitors work as an open circuit. The resistor was needed to complete the circuit and allow the signal to be processed by the A/D converter circuit.

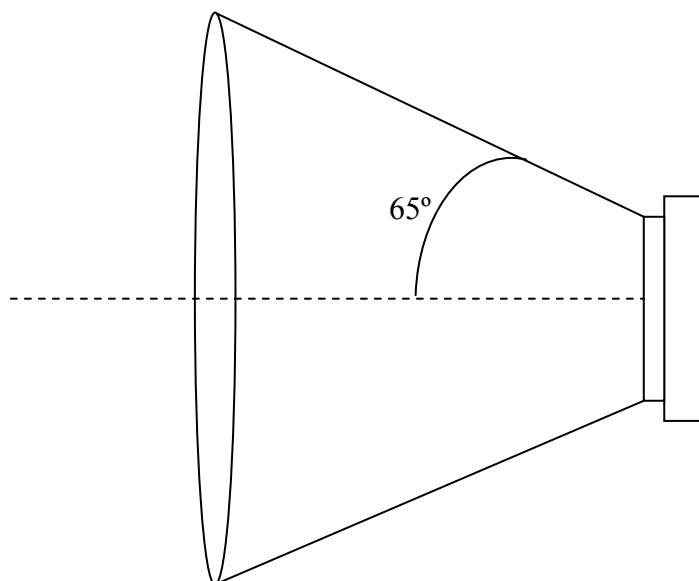


Figure 21: Directionality of Sensor

After the circuit boards were purchased for the sensors, further testing was needed to determine the consistency across the sensors and the circuits. Each circuit was tested using the same sensor and a single pulse from the transmitter. In these trials, the length of the pulse was

measured and compared to the results from the other circuits. These trials were repeated for three different angles. The circuits were tested when the transmitter was broadside (90°), left angle of broadside and right angle of broadside. The pulse lengths were consistent for each angle, yet varied some from angle to angle. The complete results for the broadside test are included in Table 5. The results for the other two angles are enclosed in the appendices. When the source was at an angle of 116.6° , the pulse width averaged about 0.0424 s. On the other side of broadside, at an angle of 68.2° , the pulse width averaged about 0.031 s. It was observed that the pulse width was the longest when the source was broadside of the receiver. The pulse width decreased as the angle from broadside increased. The other result of this experiment was the realization that the time constant within the processing circuit needed to be changed. The pulse width was much too long to achieve the desired frequency of 500 Hz. This frequency would require a maximum pulse width of 0.001 s. In order to reduce the observed pulse width in these experiments, a smaller resistor was required to change the time constant.

Table 5: Pulse Width (sec) of Single Pulse

DOA 90

	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10	Average
Circuit 1	0.045	0.044	0.044	0.044	0.043	0.044	0.044	0.044	0.044	0.044	0.044
Circuit 2	0.046	0.047	0.047	0.046	0.044	0.046	0.045	0.046	0.047	0.047	0.0461
Circuit 3	0.044	0.043	0.044	0.044	0.044	0.045	0.044	0.044	0.044	0.044	0.044
Circuit 4	0.044	0.043	0.044	0.043	0.044	0.044	0.043	0.044	0.044	0.044	0.0437
Circuit 5	0.045	0.045	0.044	0.045	0.045	0.045	0.045	0.044	0.07	0.045	0.0448
Circuit 6	0.046	0.045	0.046	0.046	0.045	0.046	0.047	0.046	0.046	0.046	0.0459
										Overall average	0.0447

4.2 Initial array testing

The initial array was constructed with the sensors and circuits on breadboards. In the initial tests a single pulse from the Devantech transmitter/receiver was sent to the array. The results from the five sensors were compared and the pulse width and time delay were examined.

Of the five sensors, three were mounted on breadboards and the other two were hard wired. The difference between the two implementation methods was noticeable in the widths of the pulses. The circuits on the breadboards had shorter pulse widths than those that had been hardwired. As seen in Figure 13, there was no time delay between the different sensor outputs when the signal was broadside to the array. Each sensor began receiving the signal at the same time. The only variation was the length of the pulse.

The array was tested using three different directions of arrival. The initial test had the receiver at broadside. There were two subsequent tests, one with the receiver closer to the right hand side of the array and the other closer to the left side of the array. In both of these cases, a delay can be seen between the pulses (Figure 22). The only problem in these cases was that not all of the sensors detected the pulse, which resulted in a zero output.

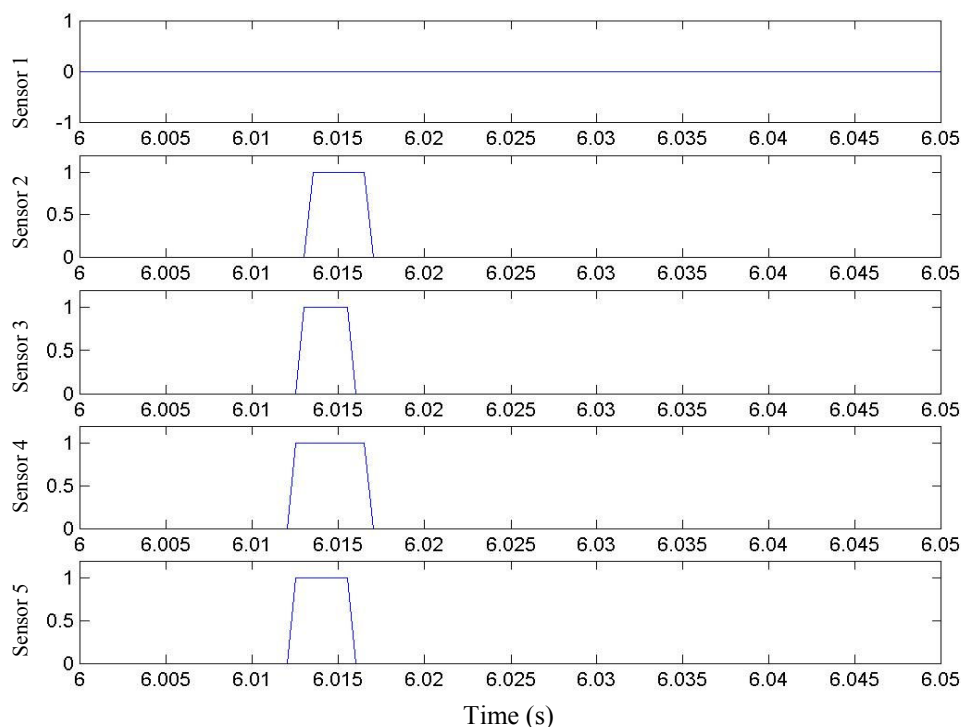


Figure 22: Single pulse (left side of array)

The pulse hit the fifth sensor slightly before it hit the other sensors when the receiver was close to the left end of the array. Similarly, there was a slight delay when the receiver was on the other side of the array (Figure 23). Due to the size of the angle, the sensors that were furthest from the receiver did not always detect the pulse. These tests illustrate that a time delay in the received signal occurred when the source was not broadside to the array.

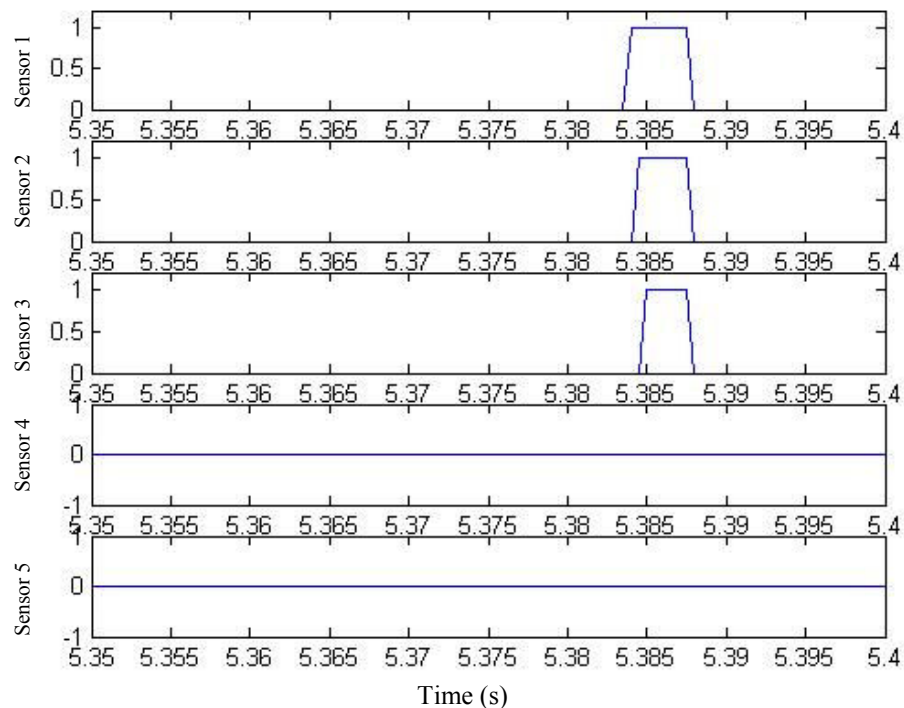


Figure 23: Single Pulse (right side of array)

4.3 Interim array testing

The array was tested with all five sensors prior to the building of the final platform. For these experiments, the sensors were mounted on stands using alligator clips. These stands were not very stable, and introduced an additional source of error. Instead of using a pulse, a continuous 500 Hz signal was used. This signal was composed of an eight-cycle burst of 40 kHz every 2 ms. This was accomplished using a function generator, which had a controllable duty

cycle in connection with a PIC processor that produced the 40 kHz signal (see Appendix C). A new transmitter was used for the periodic trials because the aforementioned signal could not be produced using the Devantech receiver/transmitter.

The array was tested initially with the source broadside to the array. The initial tests collected 10 seconds worth of data for each of the ten trials. The data was collected through the use of the Quanser interface within MATLAB (Figure 24) [Quanser, Inc., 2005]. The interface allowed the digital signal to be read into the computer for processing. The Quanser interface provided the opportunity to scope the data as it was being collected.

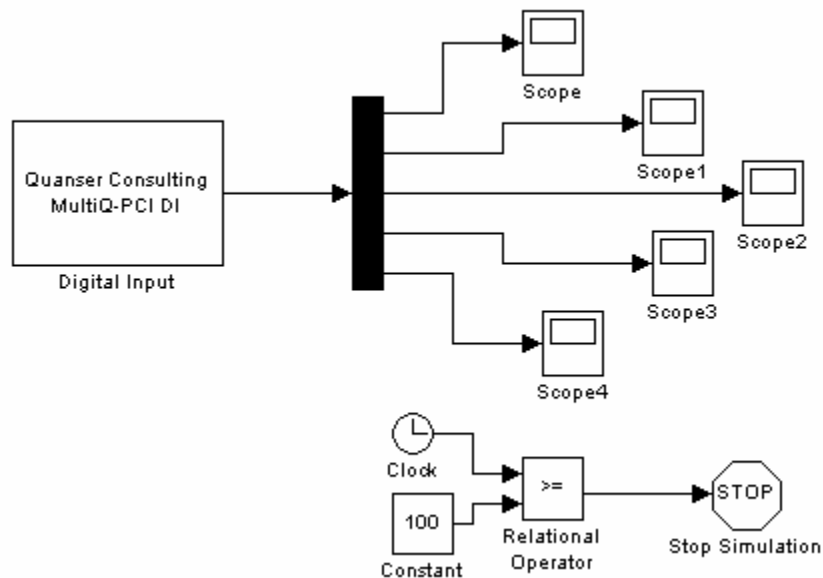


Figure 24: Quanser Model

Once the data was collected for each trial, the FFT was performed. Within the FFT, the number of data points was divided by 100 to determine the size of the snapshot, a subset of the data. This was later changed to 50 to increase the number of points within the snapshots. At this time the data matrix produced from the FFT was a 5 X 10 matrix. There were ten snapshots for each of the five sensors. This data matrix produced by the FFT contains the magnitude and angle

of the maximum of the FFT at the desired frequency is used by MuSiC for the final processing. Trials were also performed for both angles that had been used previously in the circuit testing (68.2° and 116.6°).

The FFT was recalculated after the code was modified as previously described in Section 1.4. An example of this modification was using a constant number of data points within the snapshots instead of a portion of the number of data points collected. The final calculations used snapshots of 1024 data points. The new method looked at each sensor individually rather than assuming that they all had maximums at the same point. The MuSiC algorithm was run with the new FFT data. Prior to making the changes, the MuSiC algorithm would often return an estimation of 90° , with no relation to the actual angle. This problem did not occur as often after changes were made to the method of creating in the data matrix in the FFT. This avoided the issue of obtaining erroneous data when the first sensor did not detect the signal.

The interim array was also tested to determine the range in which all five sensors would detect the signal from the source. For this experiment, data was taken at a range of 10 ft from the receivers. The source was moved along a line at this distance from the receivers. The horizontal distance was measured from the centerline. The distance was varied from five feet to the left of centerline to four feet to the right of centerline at an increment of one foot. As stated before, left and right are referring to looking at the source from the perspective of the array.

The main purpose of these experiments was to find distances at which one or more of the sensors did not detect the source at all. This did not prevent the FFT or MuSiC from being performed, but decreased the accuracy due to less information. The FFT was later changed to account for cases in which the first sensor did not detect the signal at all. The source needed to be close to broadside for all five sensors to detect it at a range of ten feet. As the distance from

90° increased, the number of sensors that detected the signal decreased. The sensors that received the signal are included in Table 6.

Table 6: Range of Array Sensors

	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5
68.2° (4ft)	X	X	X		
73.3° (3ft)	X	X	X	X	
78.7° (2ft)	X	X	X	X	
84.3° (1ft)	X	X	X	X	X
90.0° (0ft)	X	X	X	X	X
95.7° (-1ft)			X	X	X
101.3° (-2ft)			X	X	X
106.7° (-3ft)		X	X	X	X
111.8° (-4ft)			X	X	X
116.6° (-5ft)			X	X	X

4.4 Field of View Testing

The array was also tested at six different angles that were within fields of view for all five sensors. For each of these angles five sets of ten seconds worth of data was acquired. An internal clock within the Quanser interface (Figure 24) regulated the time. In the FFT calculations, the size of the snapshots was changed so that it was the same size for all of the trials. There were two different constants that were examined: 512 and 1024. These values were chosen due to the fact that they were multiples of two, which produce better results in FFT performance. The number of snapshots was also varied to determine whether that could improve the results. There were four different combinations observed: ten and fifteen sets of 1024 points and ten and twenty sets of 512 points. Within these different combinations, increasing the number of sets only affected the magnitudes of the MUSIC results.

The other objective of this testing was to determine the precision of the array when calculating the direction of arrival. Therefore small increments in angles were used to observe whether the array could make a determination between the different angles. For the most part it

was difficult to distinguish between two angles that were close to each other. The difference in angle was not large enough to create a significant difference in the results. The minimum difference in angles detected was about 5° . Either there was a DC noise component which resulted in a broadside result (the default of MuSiC) or there was too much noise and no distinct peak.

4.5 Final Platform Stationary Testing

The linear array was tested again after the final platform had been built (see Figure 14). This provided a more stable platform than what had been used in the interim. There was less possibility of the sensors to come loose and move from their intended position. During this testing, the amount of data obtained was also increased. Instead of collecting ten seconds worth of data, one hundred seconds of data was collected during every trial. This allowed an increase of the number of snapshots. Increasing the number of snapshots should reduce and cancel out more of the noise that was apparent in the array.

As in prior experiments, data was collected from three different angles (83.8° , 90° and 95.0°). For each angle, five groups of data were collected. Again, the data was divided into snapshots. The FFT was calculated using both a snapshot size of 1024 points as well as 512 points. For the data, 150 and 300 snapshots (respectively) were used. For the final experiments, 150 snapshots of 1024 data points were used. Increasing the number of snapshots within the FFT code removed the DC bias within the MuSiC estimation. For both of the angles on either side of broadside, the MuSiC estimation resulted in an angle on the expected side of broadside. When the source was located at 83.8° , the estimated angle from MuSiC was about 80° in the best estimation (Figure 25). Within the five trials, the estimated angle varied from 70° to 80° .

However, the estimation for the left angle was not as precise. The actual angle was 95.0° while the estimated angle was about 110° in the best estimate (Figure 26). Most of the time, the estimation was not accurate at all.

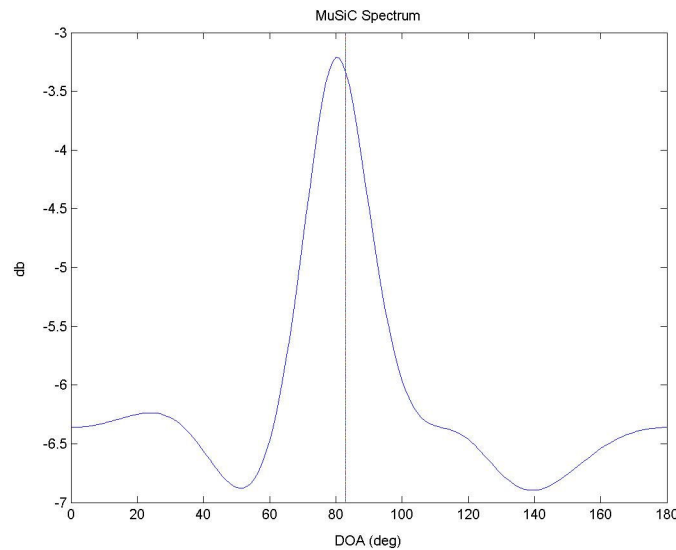


Figure 25: MuSiC Results for Right Source – Trial 2

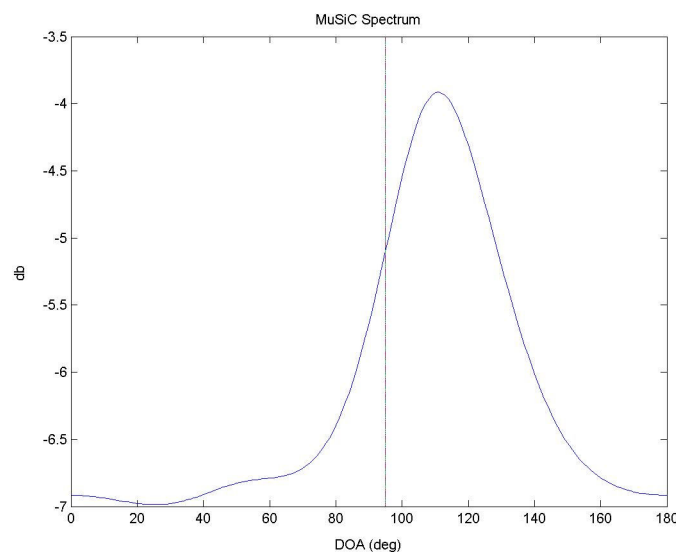


Figure 26: MuSiC Results for Left Source – Trial 1

However, it was difficult to repeat good results. Throughout the entire project, good results were difficult to obtain. Some of the factors included the size of the aperture, the size of

the snapshots and noise within the array. A minimum of two sensors is needed to estimate the direction of arrival. For this project, only five sensors were used. This is not much larger than the minimum required. The MuSiC algorithm should have canceled out much of the noise. The noise is assumed to be random, and should be averaged out when enough data is used. However, the ability of MuSiC to average out noise is limited for a small aperture.

4.6 Reconfiguration Testing

The final physical experiments examined the effect of changing the geometry on the estimation of the angle. Within these experiments, similar procedures were followed. Five trials were performed for each individual situation. Three different geometries were used on either side of the array. The data was collected in the same manner as described for the simulations in Section 3.3.

While collecting the data, it was observed that changing the geometry allowed the sensors farthest from the source in the linear array to detect more of the signal. This was in part due to the fact that both the source and the sensors were directional in the actual experiments. When the array was moved, the sensors were placed within the cone of the signal that was transmitted by the source. The data acquisition using MATLAB provided the opportunity to scope and observe the data as it was being collected. While the scopes did not show all of the information that was collected, they provided a good representation of the data.

In order to process the data from these experiments, the MuSiC algorithm had to be altered to change the assumed geometry of the array. In order to change the algorithm, the same modifications were made to the array geometry as in the simulations. The angle of change also needed to be defined within the MuSiC script.

Once the MuSiC algorithm was modified to accommodate the new geometries, the data was processed. The FFT was performed for all six configurations of the array. The data was then sent to the MuSiC algorithm and graphed for DOA estimation. The results of the DOA estimation were compared with those from the linear array. In some cases, the linear array performed better and in others, the reconfigured array performed better. When the left end of the array was 30° from the horizontal, there was no improvement in the estimation of the DOA (Figure 27). Though the estimations resulted in clear peaks, none of them were near the actual DOA of the source. However, when the left end of the array was at 45° and 60° from the horizontal, better results were obtained (Figure 28 and Figure 29). Both of these were much more consistent than the other geometry or the linear array.

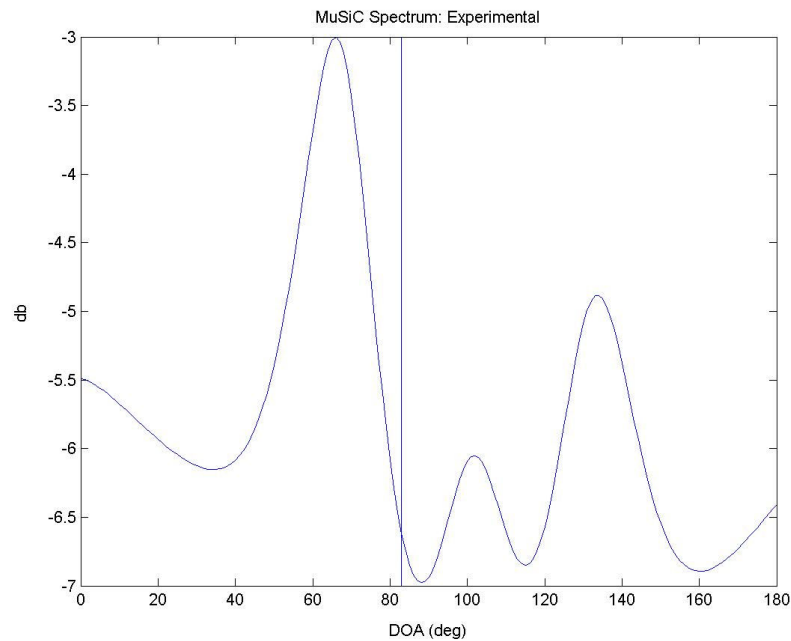


Figure 27: MuSiC results for Right Source ($\theta_a=30^\circ$) – Trial 4

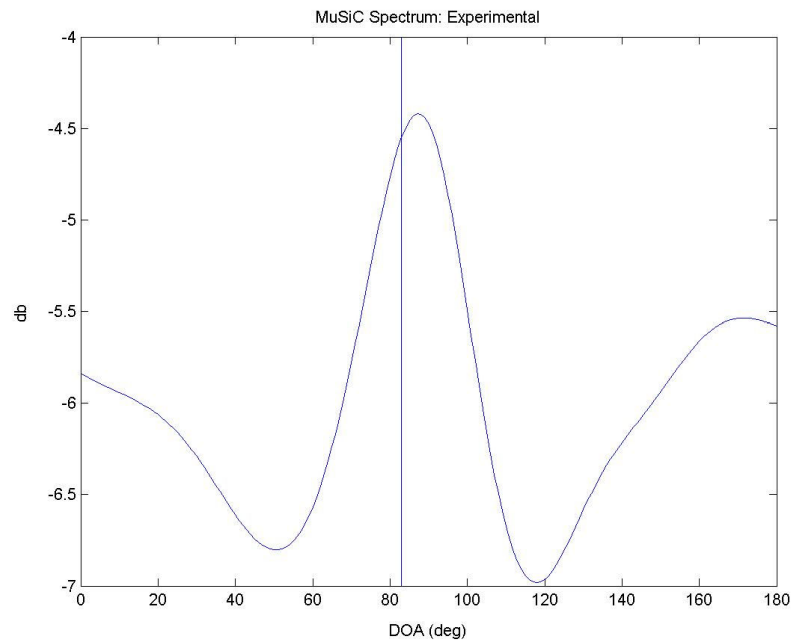


Figure 28: MuSiC Results for Right Source ($\theta_a=45^\circ$) – Trial 2

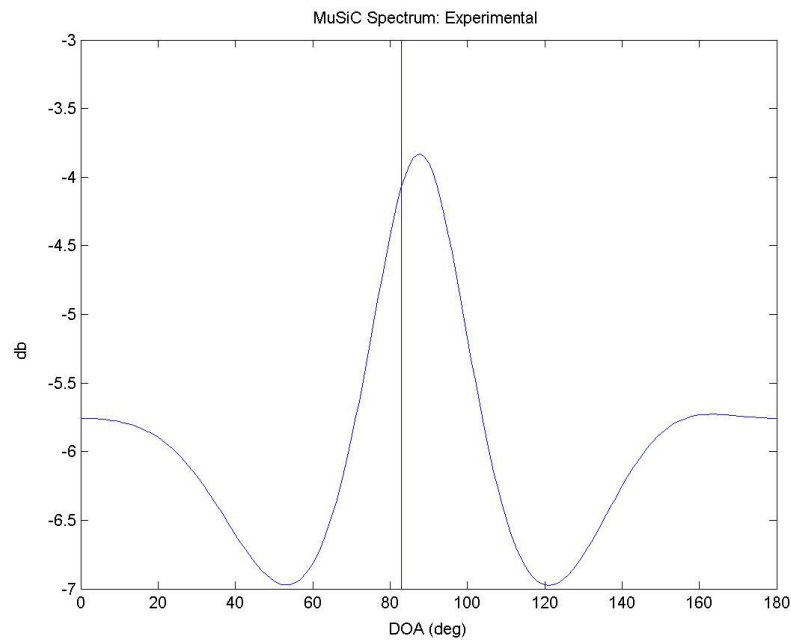


Figure 29: MuSiC Results for Right Source ($\theta_a=60^\circ$) – Trial 2

When the right end of the array was changed, it improved the estimation of the DOA. When the actual angle was 95° , the linear array estimated the angle to be about 110° . The best estimation for an array angle of 30° was also about 110° (Figure 30). This was not an improvement over the linear array. However, when the array angle was increased to 45° the

DOA estimation ranged between 90° and 115° , with an average of about 100° (Figure 31). This was the most consistent configuration of the array. When the angle of the array was 60° , the best estimation was 95° (Figure 32). Since it was inconsistent, the reconfiguration angle of 45° was a greater improvement over the linear array. Both of these geometries performed better than the linear array.

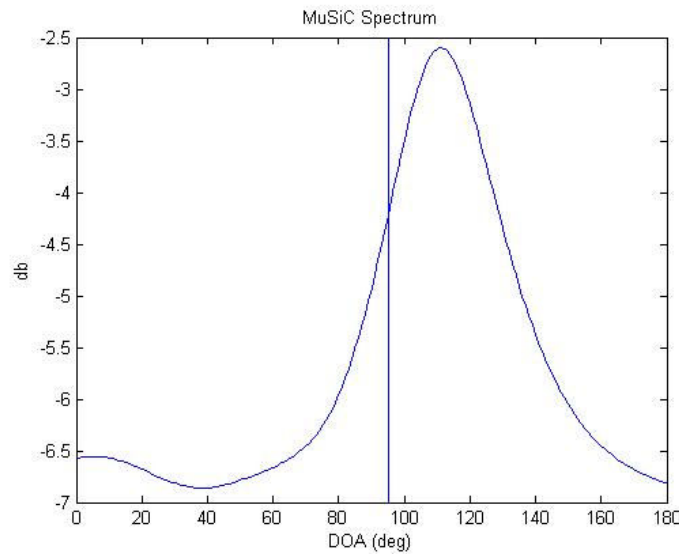


Figure 30: MuSiC Results from Left Source ($\theta_a=30^\circ$) – Trial 4

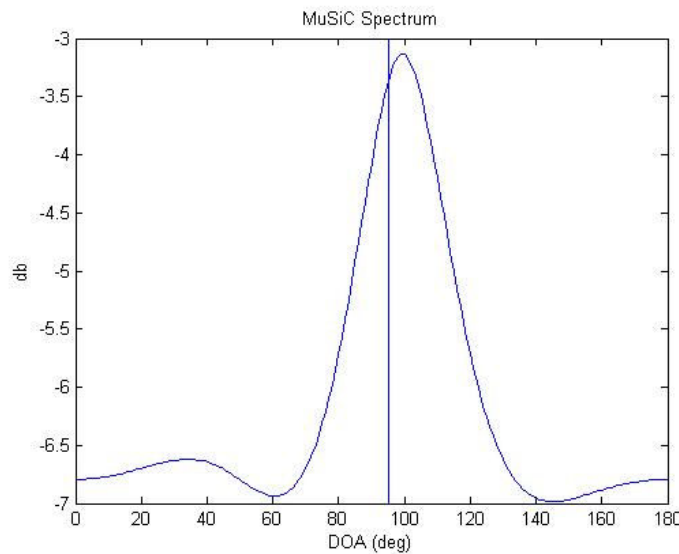


Figure 31: MuSiC Results from Left Source ($\theta_a=45^\circ$) – Trial 1

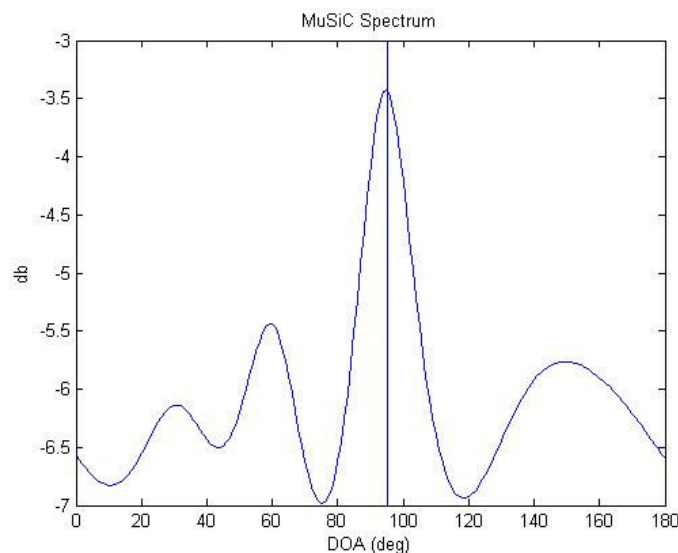


Figure 32: MuSiC Results for Left Source ($\theta_a=60^\circ$) – Trial 1

5.0 Simulations vs. Experiments

The experimental results varied greatly from the simulation results. One difference between the experiments and the simulations was that the simulations assumed that the sensors were omni directional. The actual sensors that were used were directional. They could not detect a signal from any angle. The source had to be within a cone of the face of the sensor in order for the signal to be detected. This reality in the experiments provided another reason for reconfiguration that had not been present before. The directionality of both the receivers and the transmitter required that the array be reconfigurable in order to increase the range of angles that could be detected.

Another source of variation was the noise in the system that could not be modeled within the simulations. There was unforeseen noise within the circuitry used to obtain the data. The simulations assume an environment that has random noise. The noise in the simulations was averaged out by the MuSiC algorithm, which led to inaccurate prediction of the performance of the array.

The size of the aperture also contributed to the noise in the system and the lack of precision in the experimental results. Only five sensors were used in the experiments, as well as the simulations. The simulations, however, assumed that everything precisely fit the defined parameters. The minimum number of sensors to determine direction of arrival is two. This project added three additional sensors to the minimum. This did not provide enough information to cancel out all of the noise within the system. MuSiC is not as effective if the aperture is small. If the size of the aperture was doubled, better results would probably be obtained. This noise led to the inconsistencies within the experimental data. It made results difficult to repeat due to the difference in the data obtained during each trial. There was little similarity in the amount of data that was received by each sensor. It often would change drastically between trials.

6.0 Conclusions

The main goal of this research was to modify the MuSiC algorithm to accommodate the use of a reconfigurable array in the estimation of DOA. Additionally, a physical array was to be built in order to test the original algorithm and its modifications. The assumed geometries in the algorithm were modified throughout the project to incorporate the ability to change the geometry of the array. The final platform was built and estimation of the DOA was accomplished with this platform. The other problem of interest was whether or not reconfiguring the array would improve this estimation at all. The tests that were performed demonstrated that DOA estimation could be improved when the geometry was changed from a linear array. In conclusion, MuSiC algorithm was successfully modified to estimate the DOA when the array was reconfigured.

7.0 Further Research

There were many questions that were left unanswered and new questions that became apparent through the course of research. Due to time constraints, the far field source and near field scenario was not examined in the lab. In further research, more sensors should be added to the array to create a larger aperture. This could reduce some of the problems that were inherent with the small aperture used. The platform was built to allow for computer control of the geometry of the array. This would consist of writing a simple program to control the servomotors.

Further research would be needed in order to implement these ideas in a real world application. In order to use reconfigurable arrays for harbor detection, experiments and simulations would have to be done underwater. The change in medium contributes more factors to the problem. In water, the speed of sound varies as depth and temperature change. The variations in propagation speed would need to be taken into account when designing and developing a system for underwater use.

Bibliography

- Allen, J.C. *Direction Finding Database and Techniques*: Technical Note 1657. San Diego: Naval Ocean Systems Center, 1991.
- Coe, Gerald. *Robot Electronics: Advanced Electronics to Robot Builders*. Jan 2001. Devantech Ltd. 17 April 2005 <<http://www.robot-electronics.co.uk>>
- Erdemli, Y.E., et al. "Frequency-selective surfaces to enhance performance of broad-band reconfigurable arrays." Antennas and Propagation, IEEE Transactions on. Volume: 50, Issue: 12, Dec. 2002: 1716 - 1724.
- Hassab, Joseph C. *Underwater Signal and Data Processing*. Boca Raton: CRC Press, 1989.
- Horton, C. W. *Signal Processing of Underwater Acoustic Waves*. Washington: US Government Printing Office, 1969.
- Nielsen, Richard O. *Sonar Signal Processing*. Boston: Artech House, 1991.
- Ocean, Atmosphere, and Space. 23 Oct 2003. Office of Naval Research. 20 Jan 2004 <http://www.onr.navy.mil/sci_tech/ocean/>
- Quanser: Innovate. Educate*. Quanser, Inc. 17 April 2005 <<http://www.quanser.com>>
- Simons, R.N., Donghoon Chun and L.P.B. Katehi. "Reconfigurable array antenna using microelectromechanical systems (MEMS) actuators." Antennas and Propagation Society International Symposium, 2001. IEEE. Volume: 3, 8-13 July 2001: 674 - 677 Vol.3.
- Urick, Robert J. *Principles of Underwater Sound*. New York: McGraw-Hill, 1975.
- Walker, Ashley. *Acoustics & Vibrations: Sonar R&D: Underwater*. Edge Consulting Group, LLC, 1994-2003. 20 Jan 2004 <<http://www.ecgcorp.com/velav/vldocs/sonar/h2o.html>>
- Weisstein, Eric W. "Fast Fourier Transform." From MathWorld – A Wolfram Web Resource. 14 April 2005 <<http://mathworld.wolfram.com/FastFourierTransform.html>>
- Weisstein, Eric W. "Sampling Theorem." From MathWorld – A Wolfram Web Resource. 14 April 2005 <<http://mathworld.wolfram.com/SamplingTheorem.html>>
- Wikipedia: The Free Encyclopedia*. "Discrete Fourier Transform." 9 April 2005. Free Software Foundation, Inc. 14 Apr 2005 <http://en.wikipedia.org/wiki/Discrete_Fourier_transform>
- Ziomek, Lawrence J. *Underwater Acoustics: A Linear Systems Theory Approach*. Orlando: Academic Press, Inc, 1985.

Appendix A: Glossary of Terms

Beam forming: sum multiple elements in an array to create a narrow beam to detect targets

c: sound propagation speed (343 m/s for air)

d: distance between sensors

DOA: Direction of arrival, the angle a source is at

Duty cycle: ratio of high voltage to low voltage time in a square wave signal

f: frequency of wave

FFT: Fast Fourier Transform, transforms data from the time domain to the frequency domain

Frequency domain: An alternate way to depict a set of data, related to the frequency of the signal rather than time

Function generator: a device that can produce various patterns of voltages at a variety of frequencies and amplitudes, including square waves, sine waves and triangle waves

Logic level gate: a gate that converts a signal from one logic level to the other (i.e. 0 to 1)

MATLAB: a computing environment that incorporates computation, programming and graphical simulations

MuSiC: Multiple Signal Classification, an algorithm used to estimate direction of arrival

PIC processor: a programmable chip that is used in various applications

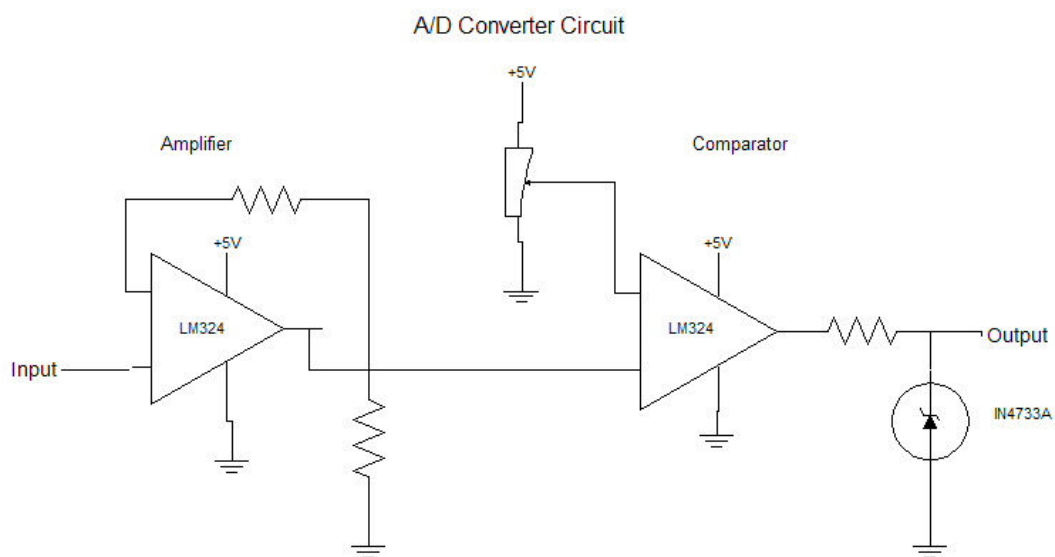
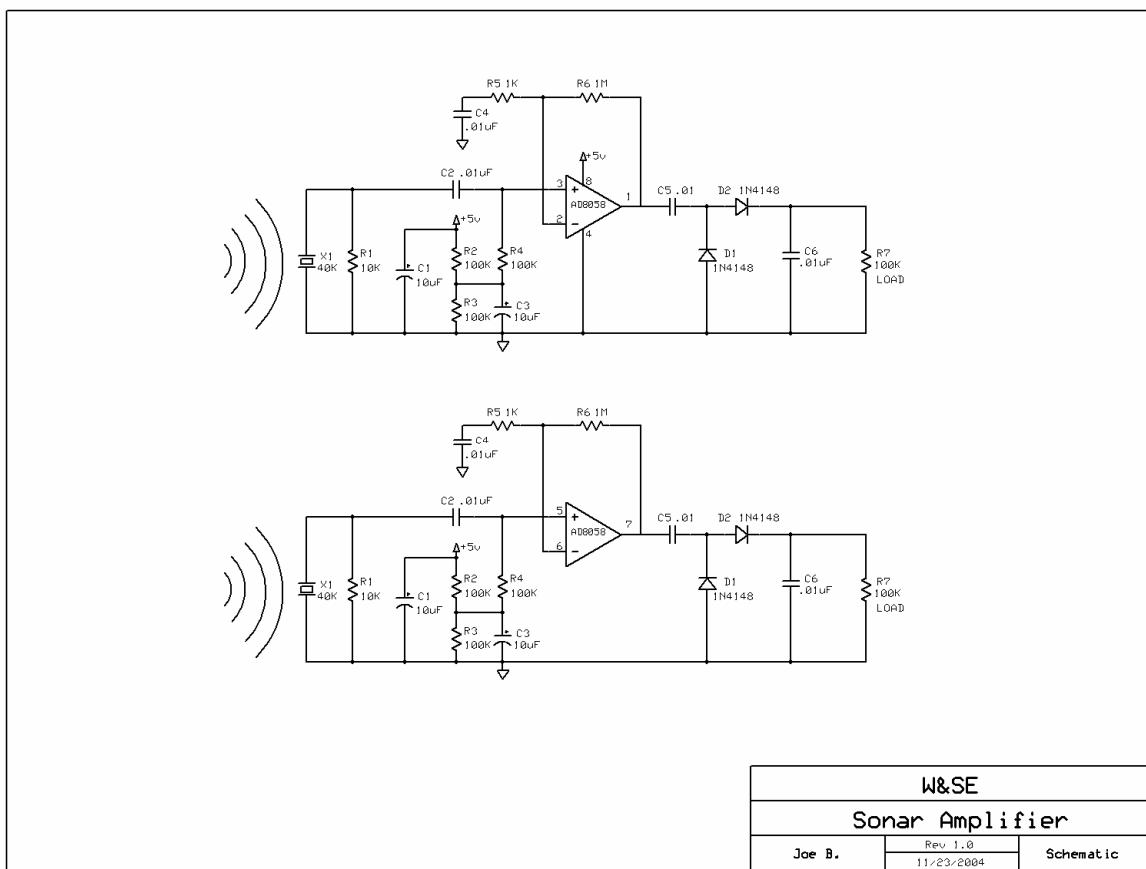
Quanser: computer interface board with a digital input/output

Script: term for a program written in MATLAB

Snapshot: a group of data within the FFT

λ : wavelength

Appendix B: Circuit Schematics



Appendix C: Program code

```
%-----
%Title:  Signal ( $y = S * a$ ) from near-field source received by sensor array
%-----
clc;clear;

f=500; %Hz
prop_speed=343; %m/s
omega=2*pi*f; %rad/sec

d=0.5*prop_speed/f;           %Spacing = Fraction of wavelength
N=5;                          %Number of sensors
theta=90*pi/180; %rad         %Angle of array
array = arraygeo(N,d,theta);  %Generates uniform array

DOAs = [180,0];               % Direction-of-arrivals of reflected source
M = size(DOAs,1);             % Number of sources

distance=2; %m
ranges = distance.*ones(size(DOAs,1),1); % Equidistant sources

S = arraypvnearf(array,DOAs,ranges,omega,prop_speed); % Array propagation
vectors
signal_strength=distance.*ones(size(DOAs,1),1); %Signal strength = 1 at the
array center
signal_strength=0.1*signal_strength;

mu=0; % Noise parameters
SNR=2000; %db
signal_arraycenter=signal_strength/distance;
sig2=signal_arraycenter^2*10^(-SNR/10); %SNR at the array center, where
signal strength = 1
sig=sqrt(sig2/2);           %sig2=sig^2+sig^2
nsnapshots=100;
T=1/3/f; %Sampling time  $T=2\pi/3/\omega$ 
omega_vec = omega*ones(size(DOAs,1),1);

for i=1:nsnapshots
    a(:,i)=signal_strength;
    X(:,i)=S*a(:,i);%+noise;
end

save signal_n_parameters omega prop_speed array d M DOAs ranges SNR X
distance
X_nf=X;save c:\MUSIC\nearfield_reflection X_nf
```

```

%-----
%Title:  Signal (y = S * a) from far-field source received by sensor array
%Sub-title:  The file signal_MDLAIC.m is very similar
%-----

clc;clear;

disp('Generating source and medium parameters...');
f=500; %Hz
prop_speed=343; %m/s
omega=2*pi*f;
omega_pv=mean(omega); %Array propagation vectors w/ same frequency

disp('Generating uniform linear array...');
d=0.5*prop_speed/f;          %Spacing = Fraction of wavelength
N=5;
theta=30*pi/180;
array = arraygeo(N,d,theta);          %Generates array geometry

disp('Generating source DOAs...');
DOAs = [95,0];          % Direction-of-arrivals of 1 source
M = size(DOAs,1);          % Number of sources

disp('Generating data matrix...');
modelingerror=0;          %1==YES or 0==NO
sig_p=0.21;mu_p=0;
sig_g=0.35;mu_g=0;
sig_pg=[sig_p;sig_g];mu_pg=[mu_p;mu_g];

nfailedsensors=0; %Failed sensors or, for block failure, number of
failrate=0.25;

S =
arraypv(omega_pv,prop_speed,array,DOAs,modelingerror,sig_pg,mu_pg,nfailedsens
ors,failrate);

          % Array propagation vectors

mu=0;
SNR=0;sig2=1^2*10^(-SNR/10); %signal strength = 1
sig=sqrt(sig2/2);          %sig2=sig^2+sig^2
nsnapshots=100;
sum=zeros(N);
T=1/3/max(f); %Sampling time T=2*pi/3/omega
for i=1:nsnapshots
    a(:,i)=ones(size(omega));
    noise=(sig*randn(N,1)+mu)+j*(sig*randn(N,1)+mu);
    X(:,i)=S*a(:,i)+noise;
end

save signal_n_parameters omega prop_speed array d M DOAs SNR X

```



```

%A tutorial on DFT:  Computation of FFT
%Spring 2005
%K. Kiriakidis
%Modified by 1/c Danica Adams

clear;close all;
load c:\MATLAB\work\Trident\Geometry_tests\Right_60\left1_60_data.mat
data=left1;                                %data matrix from sensors
sampling_int=.0005;                        %sampling time of Quanser
N=1024;                                    %number of elements in a snapshot
k=[0:1:N-1]';
nsnapshots=150;                            %used with music
for i=1:nsnapshots
    ii=(i-1)*N+1;
    y=data(ii:ii+N,:);                    %vector of N data x sensors
    y_FT=fft(y,N);                        %FFT: N frequencies x sensors
    OMEGA_0=2*pi/N;                       %frequency spacing
    f=[0:N/2-1]*OMEGA_0/2/pi/sampling_int;%frequency vector in Hz

figure(1)
subplot(1,2,1)
plot(f,abs(y_FT(N/2+1:N,1)),'.')          %magnitude of FFT at sensor no 1
xlabel('f (Hz)')
ylabel('|Y(k)|')

subplot(1,2,2)
plot(f,(180/pi)*angle(y_FT(N/2+1:N,1)),'.') %angle of FFT at sensor no 1 for
first snapshot
xlabel('f (Hz)')
ylabel('<Y(k)')

[magn_peak,index]=max(abs(y_FT(3*N/4-50:3*N/4+50,:)),[],1); %finds the
maximum of the FFT

for j=1:5
    X(j,i)=y_FT(index(j),j);              %creates matrix of maximum's
    magnitude and angle
end
end

save fft_left1.mat X

```

```

%script M-file: script.m
%-----
%Compute the MuSiC spectrum from data file
%-----

clc;clear;

%load signal_n_parameters omega prop_speed array d M DOAs SNR X;
%load nearfield_reflection; X=X+X_nf; %add near-field to far-field signal
load C:\MATLAB\work\Trident\Geometry_tests\Left_30\fft_right1.mat

N = size(X,1); %Number of sensors

nsnapshots=size(X,2);sum=zeros(N); %Number of snapshots
for i=1:nsnapshots
    sum=sum+X(:,i)*X(:,i)';
end
R_approx=sum/nsnapshots; % Estimated covariance matrix

%Performing MUSIC algorithm
f=500; %Hz
prop_speed=343; %speed of sound in air
omega=2*pi*f;
omega_music=mean(omega); %assume signals at the same frequency
d=0.5*prop_speed/f; %Spacing = Fraction of wavelength
theta=30*pi/180; %change in geometry (convert from
degrees to radians)
array = arraygeo(N,d,theta); %model geometry of array
M=1; %number of sources
Z=music(omega_music,prop_speed,R_approx,array,M,[0:180],0); %
MUSIC algorithm
[Z_max index_max]=max(Z);
theta_hat=index_max-1;

figure(2)
plot([0:180],Z) % Display result
xlabel('DOA (deg)');ylabel('db')
title(['MuSiC Spectrum: Experimental'])

function array=arraygeo(N,d,theta);
%*****
%Title: Sensor position vectors
%*****

N_half=(N-1)/2; %N must be an odd number
%right end of the array
for mm=1:3;array(:,mm)=(mm-1-N_half)*d*[cos(theta) -sin(theta) 0]';end
for mn=3:N;array(:,mn)=(mn-1-N_half)*d*[1 0 0]';end% 3Xnsensors
%left end of the array
for mm=1:3;array(:,mm)=(1+N_half-mm)*d*[1 0 0]';end
for mn=3:N;array(:,mn)=(1+N_half-mn)*d*[cos(theta) sin(theta) 0]';end
%linear array
for mm=1:N;array(:,mm)=(mm-1-N_half)*d*[1 0 0]';end

```

```

function
Z=music(omega,prop_speed,Rxx,array,M,AZarea,ELarea,mainlobe,ingain,inphase);
%*****
% Z=music(Rxx,array,M,AZarea,ELarea,mainlobe,ingain,inphase)
% estimates the MuSIC spectrum.
% N.B.: if f and c are given then use array*2*f/c
%-----
% written by Dr A.Manikas (IC)
% Modified 5.6.01 - Jason W.P. Ng
%*****
N=size(array,2); %Number of array elements

if nargin<9; g=ones(N,length(AZarea)); %NARGIN Number of function input
arguments
    else g=repc(ingain.*exp(j*inphase),length(AZarea));
end;
if nargin<8
    mainlobe=[];
end;

[MEIG,D] = eig(Rxx);
[lamda,k]=sort(diag(D)); %sorts the elements of D (eigenvalues) in ascending
order
MEIG=MEIG(:,k); %sorts the eigenvectors respectively

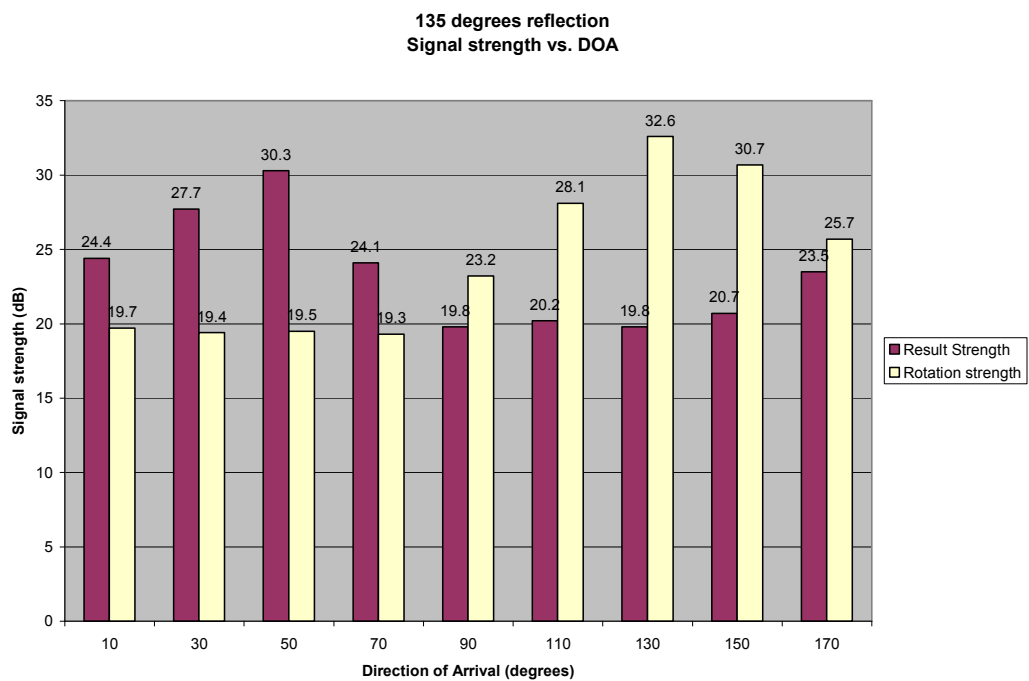
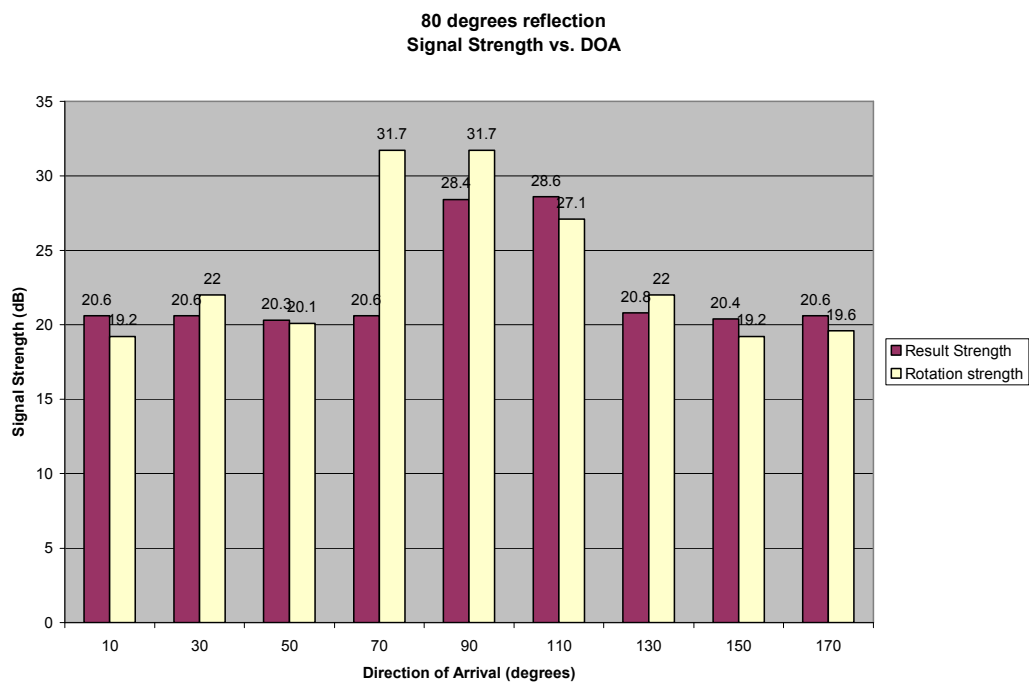
EE = MEIG(:,1:N-M); %N-M = #elements - #sources, the "noise" eigenvectors
for I=1:N-M;
    EE(:,I)=EE(:,I)/sqrt(abs(EE(:,I)'+EE(:,I))); %unit length eigenvectors
end;

y=0;
for el=ELarea;
    y=y+1;
    x=0;
    for az=AZarea;
        x=x+1;
        SOURCES=[az,el];
        S=spv(omega,prop_speed,array,SOURCES,mainlobe);
        Z(y,x)=-10*log10(real(diag(S'*EE*EE'*S))');
    end
end;
end;

```

Appendix D: Simulation Results

Rotation Simulation Results



Reconfiguration Simulations Results

Magnitude of MuSiC Spectrum for DOA 40°

Reflection angle	Linear array	Left 30°	Left 45°	Left 60°
0°	18.6 dB	19.9 dB	17.2 dB	21.3 dB
10°	18.4 dB	19.3 dB	17.1 dB	20.4 dB
20°	18.0 dB	19.1 dB	17.5 dB	19.6 dB
30°	17.1 dB	20.0 dB	18.1 dB	18.9 dB
40°	16.0 dB	22.4 dB	18.5 dB	18.5 dB
50°	15.6 dB	25.6 dB	17.8 dB	18.3 dB
60°	16.0 dB	25.4 dB	16.7 dB	18.0 dB
70°	16.8 dB	21.5 dB	16.8 dB	18.1 dB
80°	18.7 dB	19.2 dB	18.4 dB	18.6 dB
90°	21.7 dB	19.1 dB	19.1 dB	19.0 dB
100°	21.4 dB	19.3 dB	19.0 dB	19.7 dB
110°	19.8 dB	19.5 dB	19.2 dB	21.5 dB
120°	19.6 dB	21.2 dB	19.4 dB	23.3 dB
130°	20.4 dB	22.8 dB	19.4 dB	23.7 dB
140°	21.5 dB	22.4 dB	19.7 dB	22.7 dB
150°	22.3 dB	21.7 dB	20.9 dB	21.2 dB
160°	21.5 dB	22.4 dB	22.6 dB	20.8 dB
170°	19.8 dB	24.2 dB	23.2 dB	21.8 dB
180°	19.2 dB	25.8 dB	23.2 dB	22.0 dB

Estimated DOA

Reflection angle	Linear array	Left 30°	Left 45°	Left 60°
0°	39°	40°	40°	40°
10°	39°	40°	41°	40°
20°	39°	40°	41°	39°
30°	39°	40°	40°	38°
40°	40°	40°	38°	38°
50°	41°	39°	38°	38°
60°	41°	39°	38°	39°
70°	41°	39°	40°	39°
80°	40°	40°	41°	39°
90°	40°	41°	40°	38°
100°	40°	40°	38°	37°
110°	41°	40°	37°	37°
120°	42°	39°	37°	37°
130°	41°	39°	37°	37°
140°	40°	39°	37°	37°
150°	39°	39°	37°	38°
160°	39°	38°	38°	39°
170°	39°	38°	39°	40°
180°	39°	39°	40°	40°

Magnitude of MuSiC Spectrum for DOA 160°

Reflection angle	Linear array	Right 30°	Right 45°	Right 60°
0°	25.5 dB	22.4 dB	17.7 dB	20.1 dB
10°	25.2 dB	25.4 dB	19.3 dB	19.2 dB
20°	24.3 dB	25.7 dB	21.4 dB	18.2 dB
30°	23.4 dB	25.6 dB	23.0 dB	17.6 dB
40°	22.0 dB	25.1 dB	23.6 dB	17.4 dB
50°	19.8 dB	25.1 dB	22.9 dB	17.2 dB
60°	18.1 dB	22.0 dB	20.8 dB	16.7 dB
70°	17.9 dB	21.9 dB	18.8 dB	15.5 dB
80°	18.5 dB	20.4 dB	18.5 dB	14.5 dB
90°	19.1 dB	18.3 dB	18.1 dB	14.4 dB
100°	19.4 dB	17.9 dB	17.1 dB	15.1 dB
110°	18.9 dB	17.8 dB	18.2 dB	16.3 dB
120°	18.0 dB	17.6 dB	19.6 dB	18.3 dB
130°	18.6 dB	18.4 dB	18.7 dB	19.3 dB
140°	21.1 dB	18.4 dB	19.2 dB	17.6 dB
150°	24.4 dB	18.1 dB	18.8 dB	17.2 dB
160°	26.5 dB	18.8 dB	17.7 dB	17.1 dB
170°	27.5 dB	20.1 dB	17.9 dB	16.6 dB
180°	27.8 dB	21.4 dB	19.0 dB	16.4 dB

Estimated DOA

Reflection angle	Linear array	Right 30°	Right 45°	Right 60°
0°	165°	157°	158°	157°
10°	165°	160°	159°	157°
20°	164°	162°	161°	157°
30°	162°	162°	162°	157°
40°	161°	162°	163°	158°
50°	161°	162°	163°	159°
60°	162°	158°	163°	159°
70°	164°	156°	160°	159°
80°	165°	156°	157°	157°
90°	164°	159°	156°	156°
100°	163°	163°	161°	156°
110°	163°	163°	164°	158°
120°	164°	160°	165°	159°
130°	166°	158°	162°	159°
140°	168°	158°	158°	158°
150°	167°	159°	156°	156°
160°	165°	160°	158°	156°
170°	164°	161°	160°	156°
180°	164°	161°	160°	156°

Appendix E: Experimental Results

Circuit Testing Results

Pulse Width (sec) of Single Pulse

DOA 63.4

	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10	Average
Circuit 1	0.042	0.041	0.043	0.042	0.041	0.042	0.043	0.043	0.042	0.04	0.042
Circuit 2	0.044	0.043	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044
Circuit 3	0.04	0.04	0.04	0.04	0.04	0.041	0.041	0.045	0.043	0.042	0.041
Circuit 4	0.043	0.043	0.041	0.044	0.042	0.042	0.042	0.04	0.041	0.041	0.042
Circuit 5	0.043	0.042	0.042	0.043	0.043	0.043	0.043	0.041	0.043	0.044	0.043
Circuit 6	0.043	0.044	0.044	0.043	0.043	0.04	0.043	0.042	0.043	0.043	0.043
										Overall average	0.0425

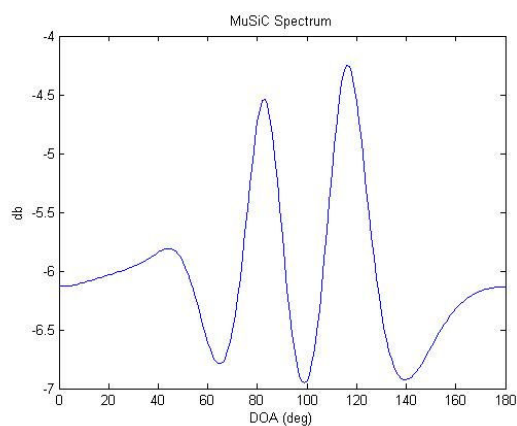
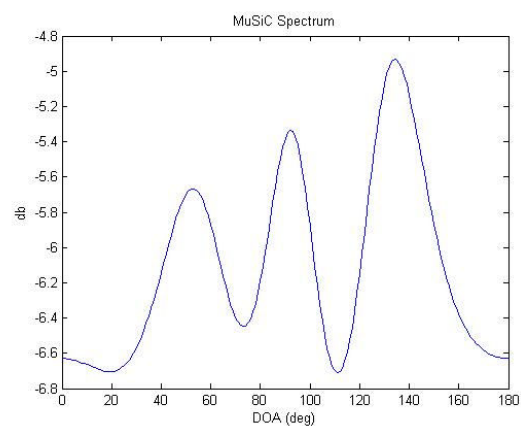
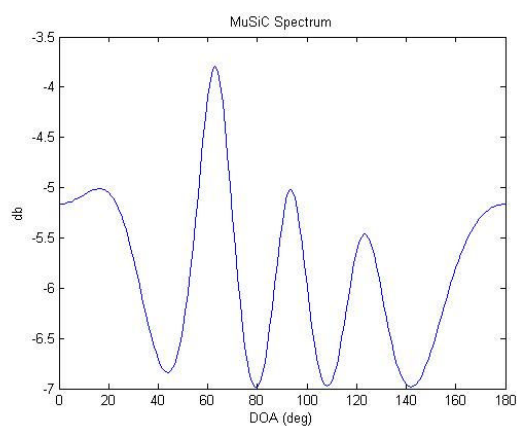
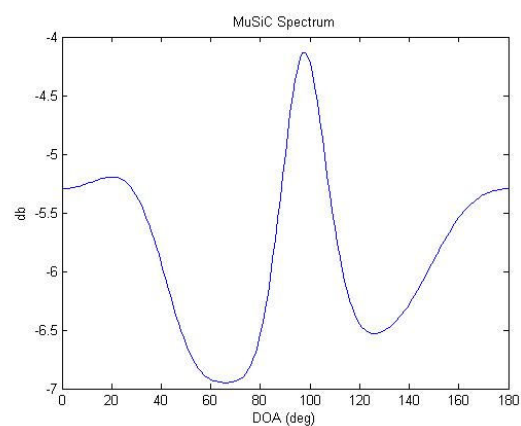
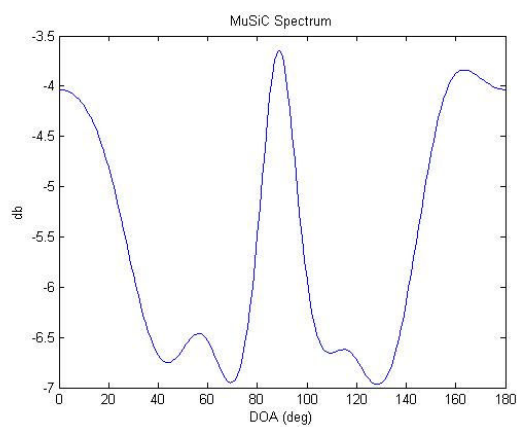
Pulse Width (sec) of Single Pulse

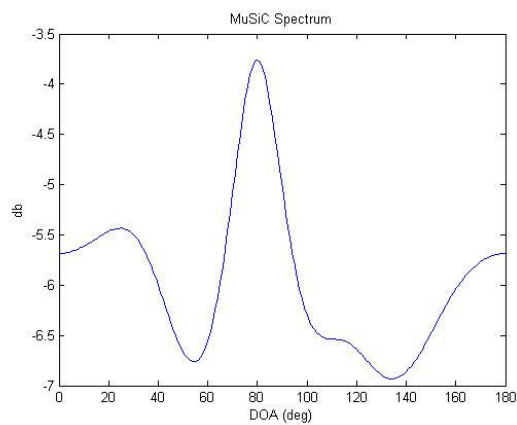
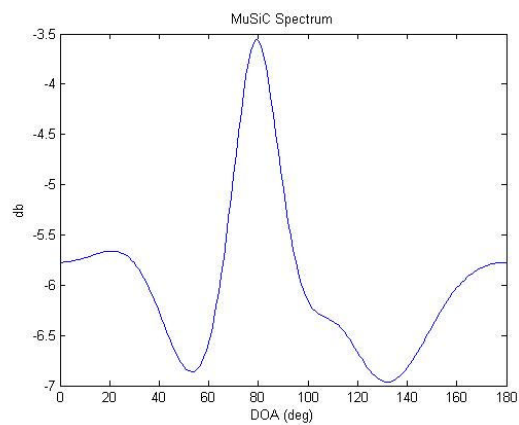
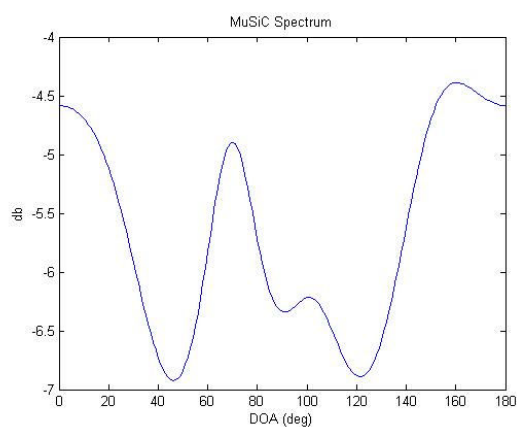
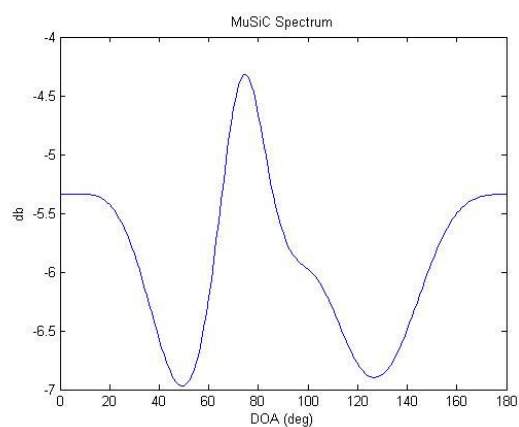
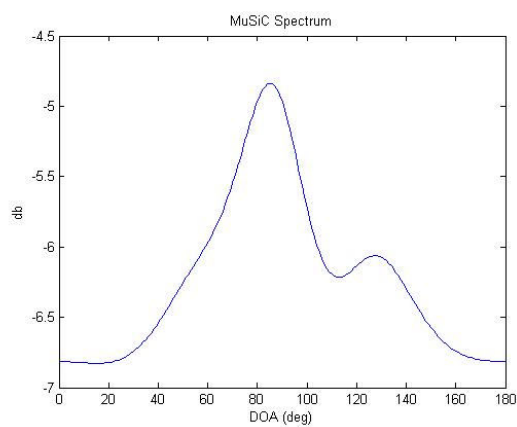
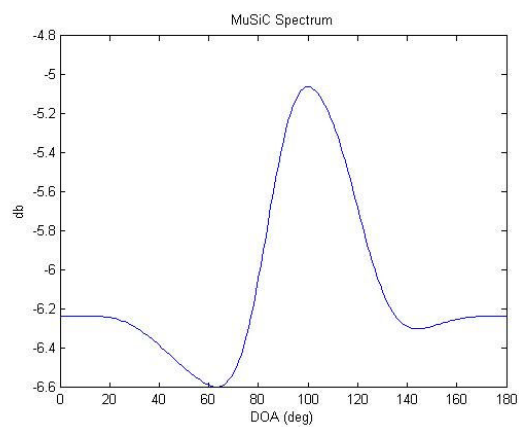
DOA 111.8

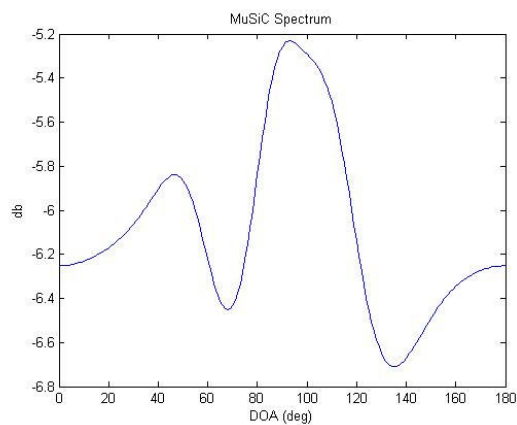
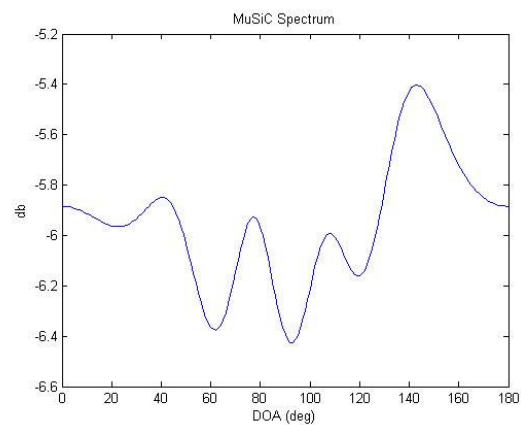
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10	Average
Circuit 1	0.035	0.036	0.038	0.033	0.036	0.035	0.037	0.035	0.038	0.036	0.036
Circuit 2	0.034	0.035	0.036	0.036	0.034	0.036	0.016	0.034	0.037	0.035	0.033
Circuit 3	0.03	0.029	0.031	0.029	0.031	0.031	0.029	0.03	0.028	0.03	0.030
Circuit 4	0.022	0.022	0.024	0.023	0.026	0.026	0.02	0.023	0.025	0.024	0.024
Circuit 5	0.019	0.023	0.02	0.035	0.034	0.033	0.032	0.034	0.035	0.035	0.03
Circuit 6	0.034	0.034	0.034	0.034	0.035	0.036	0.035	0.034	0.035	0.037	0.035
										Overall average	0.031

Linear array experimental results

MuSiC Results for DOA 90°

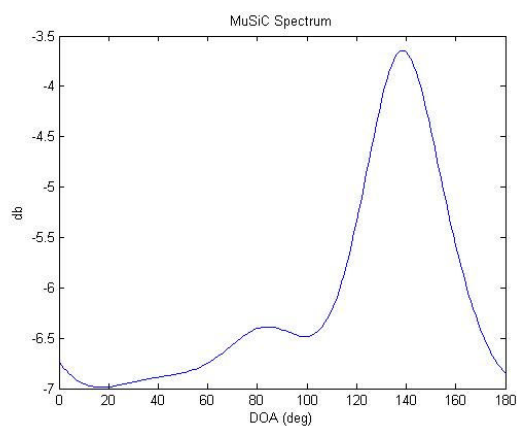
**Trial 1****Trial 2****Trial 3****Trial 4****Trial 5**

MuSiC Results for DOA 83° **Trial 1****Trial 3****Trial 4****Trial 5**MuSiC Results for DOA 95° **Trial 2****Trial 3**

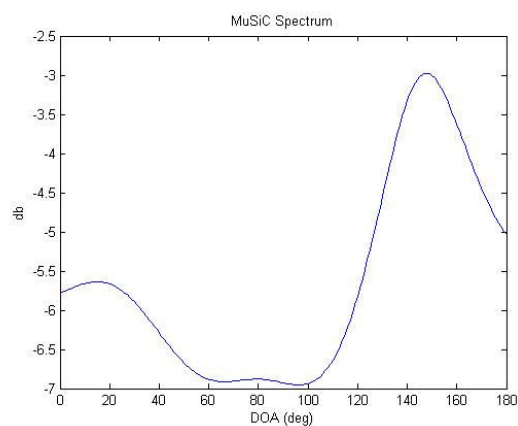
**Trial 4****Trial 5**

Reconfigured array experimental results

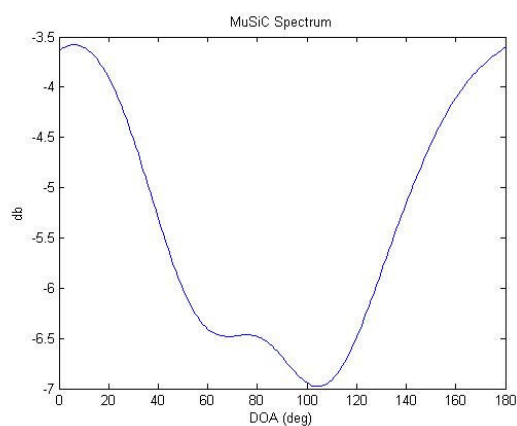
MuSiC Results for DOA 83° , Left Array Angle = 30°



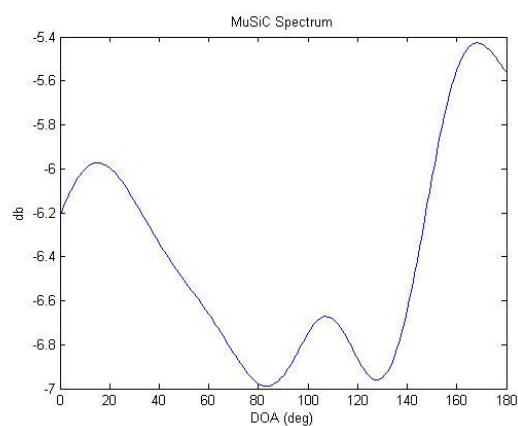
Trial 1



Trial 2

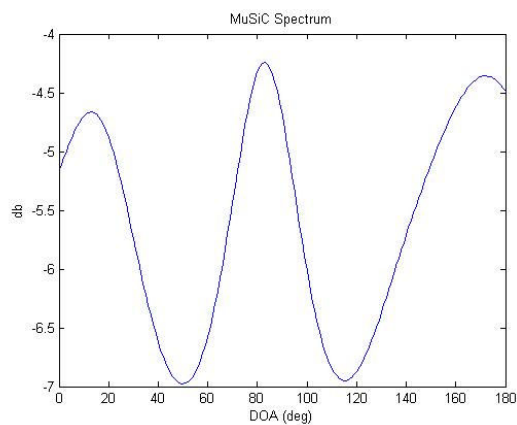


Trial 3

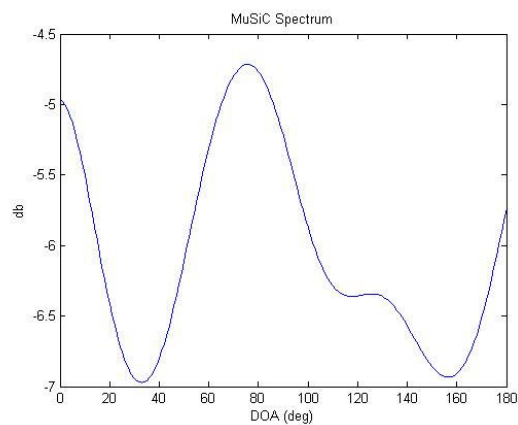


Trial 5

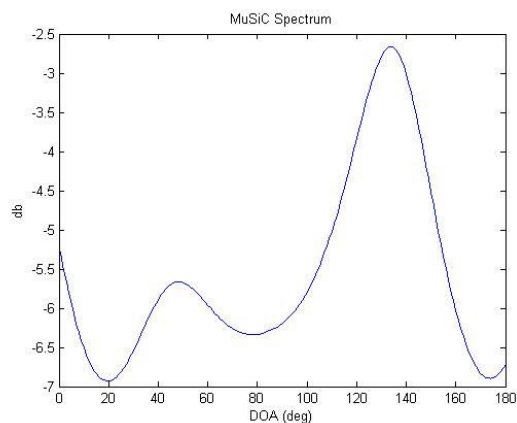
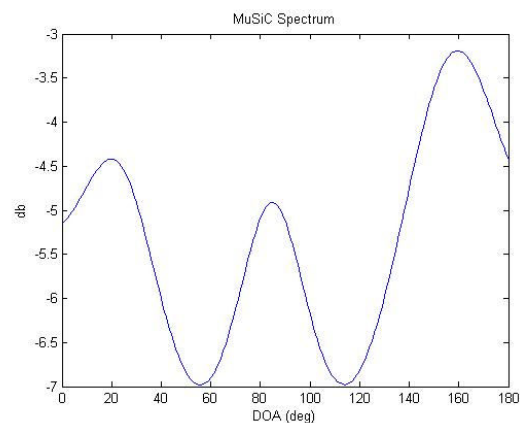
MuSiC Results for DOA 83° , Left Array Angle = 45°



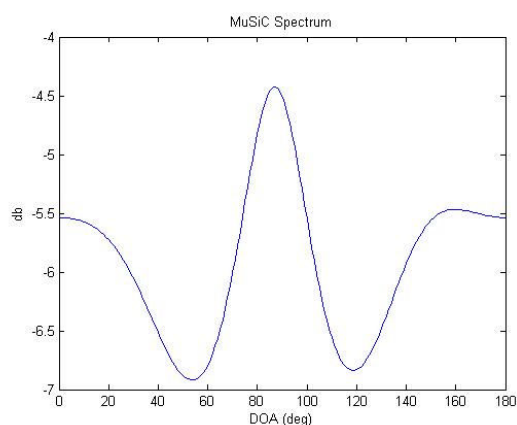
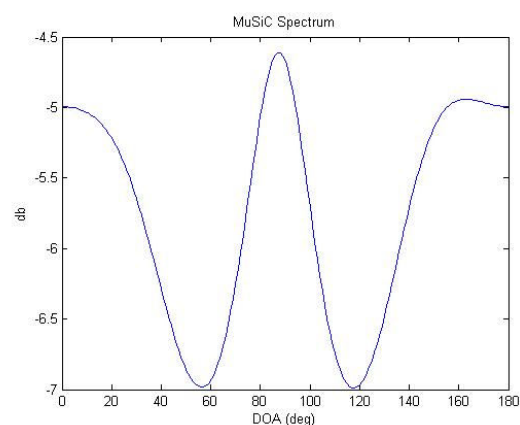
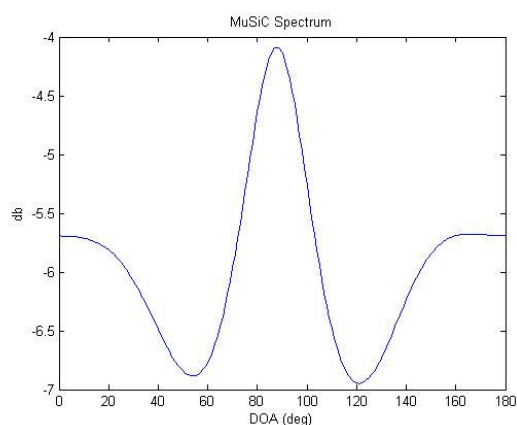
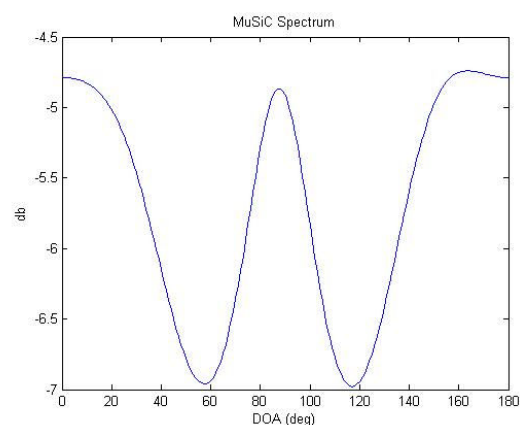
Trial 1



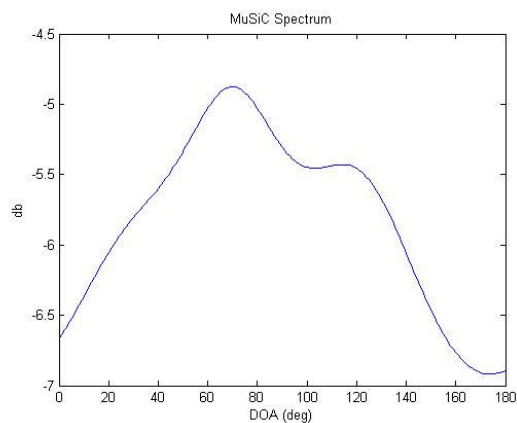
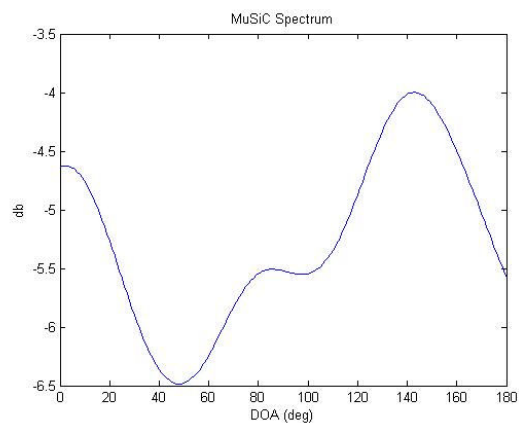
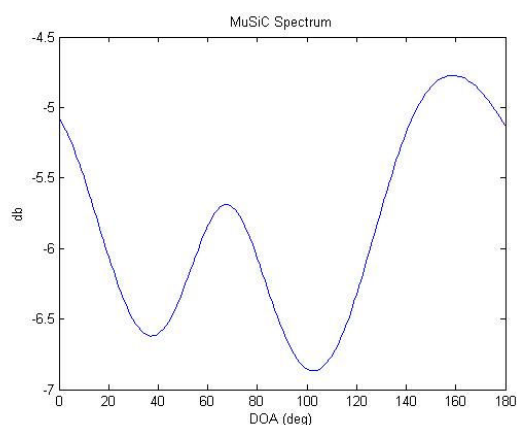
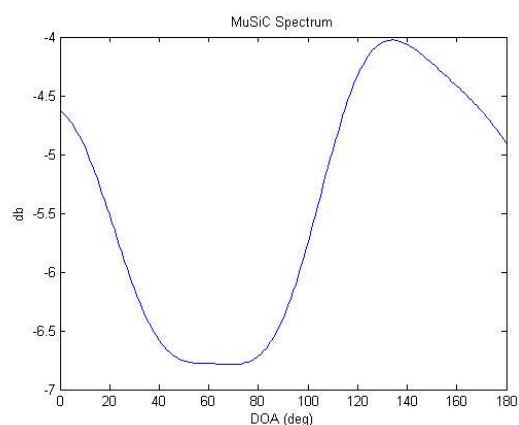
Trial 3

**Trial 4****Trial 5**

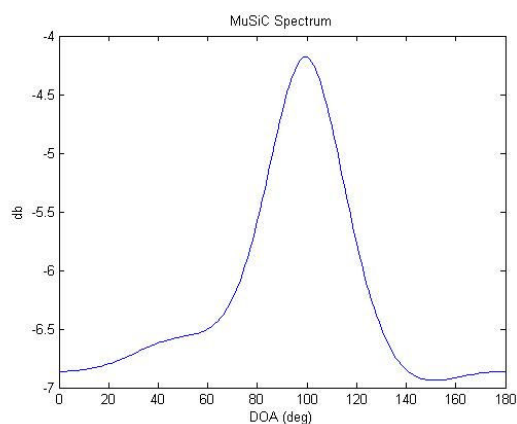
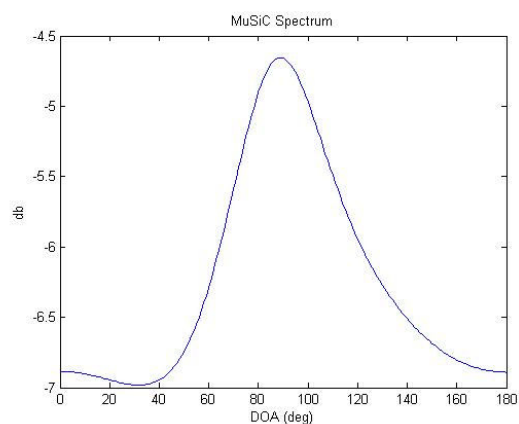
MuSiC Results for DOA 83° , Left Array Angle = 60°

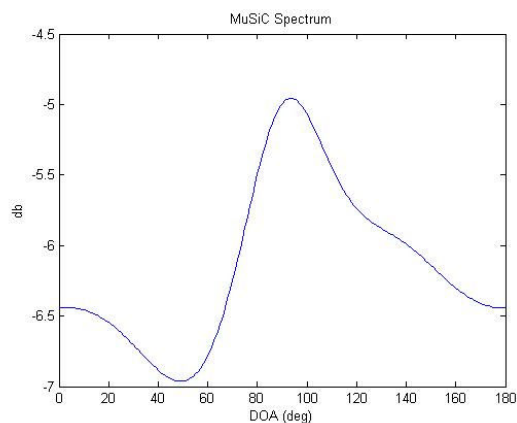
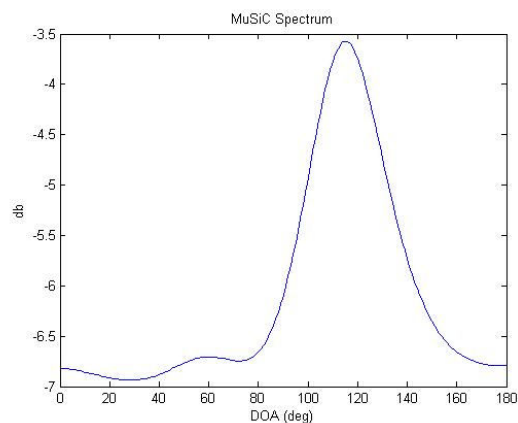
**Trial 1****Trial 3****Trial 4****Trial 5**

MuSiC Results for DOA 95°, Right Array Angle = 30°

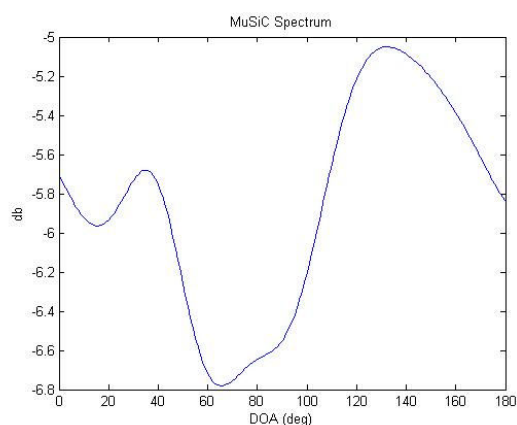
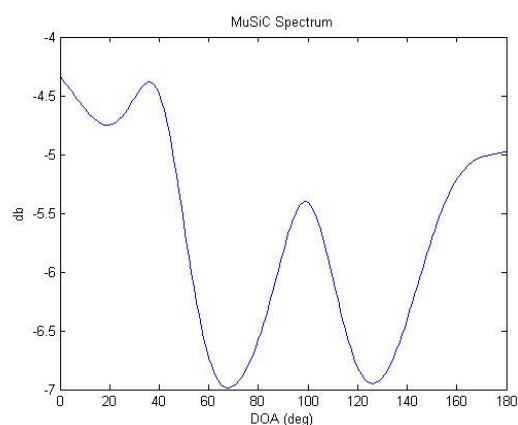
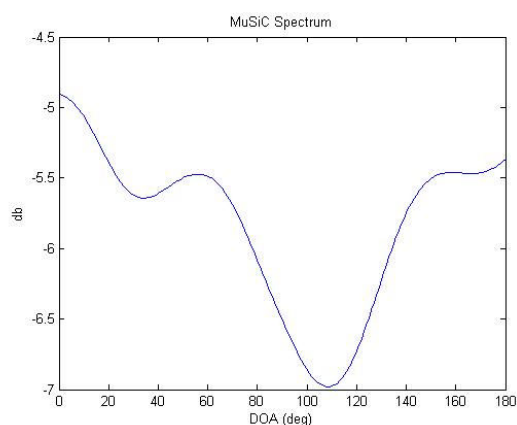
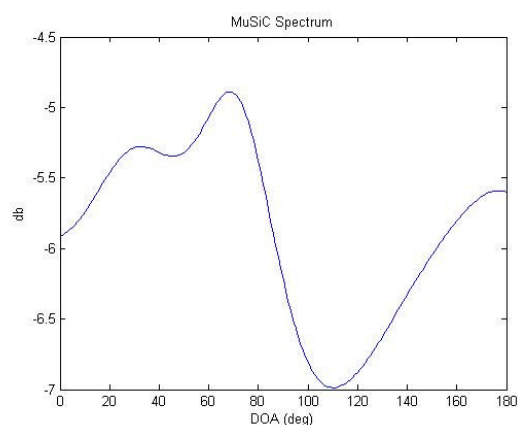
**Trial 1****Trial 2****Trial 3****Trial 5**

MuSiC Results for DOA 95°, Right Array Angle = 45°

**Trial 2****Trial 3**

**Trial 4****Trial 5**

MuSiC Results for DOA 95°, Right Array Angle = 60°

**Trial 2****Trial 3****Trial 4****Trial 5**